ELSEVIER

Available at
www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

Theoretical Computer Science 304 (2003) 315–339

Theoretical Computer Science

www.elsevier.com/locate/tcs

Shape preserving top-down tree transducers

Zoltán Fülöp*, Zsolt Gazdag

Department of Computer Science, University of Szeged, Árpád tér 2., H-6720 Szeged, Hungary

Received 17 December 2001; received in revised form 13 January 2003; accepted 21 January 2003
Communicated by A. Salomaa

Abstract

As top-down tree transducers generalize generalized sequential machines, shape preserving top-down tree transducers naturally generalize length preserving generalized sequential machines. For instance, top-down relabeling tree transducers are shape preserving top-down tree transducers. We show that a top-down tree transducer is shape preserving if and only if it is equivalent to a top-down relabeling tree transducer. We also prove that it is decidable if a top-down tree transducer is shape preserving.

1. Introduction

A generalized sequential machine (gsm) is a system $M=(Q,\Sigma,\Delta,q_0,\delta,F)$, where Q is the set of states; Σ and Δ are the input and the output alphabets, respectively; q_0 is the initial state; $F\subseteq Q$ is the set of final states; and δ, the transition function, is a mapping from $Q\times\Sigma$ to the finite subsets of $Q\times\Delta^*$. Then δ extends from $Q\times\Sigma^*$ to the finite subsets of $Q\times\Delta^*$ in a standard way and the translation defined by M is the set $\tau_M=\{(x,y)\in\Sigma^*\times\Delta^*\mid(q,y)\in\delta(q_0,x)$ for some $q\in F\}$.

In general the length of an input string $x\in\Sigma^*$ and of an output string $y\in\tau_M(x)$ is not the same, however if τ_M has this property then M is called a length preserving gsm. For instance if M is a Mealy automaton, i.e., δ maps to the subsets of $Q\times\Delta$, then M is length preserving. It is a well known result that in fact only Mealy automata are length preserving gsm's in the sense that a gsm M is length preserving if and only if it is equivalent to a Mealy automaton [1,7].

*Corresponding author. Tel.: +36-62-311-184; fax: +36-62-312-292. Research of this author was supported by the Hungarian Scientific Foundation (OTKA) under Grant T 030084.

E-mail addresses: fulop@inf.u-szeged.hu (Z. Fülöp), gazdag@inf.u-szeged.hu (Z. Gazdag).

0304-3975/03/$ - see front matter
doi:10.1016/S0304-3975(03)00140-3

In this paper we generalize this result to top-down tree transducers. While a gsm operates over strings, a top-down tree transducer works on terms (or rather trees), which are called also trees.

More exactly, a top-down tree transducer [8,2] is a system $M=(Q,\Sigma,\Delta,q_0,R)$, where Q is the set of states; Σ and Δ are the input and the output ranked alphabets, respectively, and q_0 is the initial state. Moreover, R is a finite set of (rewriting) rules of the form $q(\sigma(x_1,\ldots,x_k))\to r$, where $q\in Q$, σ is an input symbol of arity k from Σ, and r is a term over Δ which may contain also constructs as $p(x_i)$, where p is a state and $1\leqslant i\leqslant k$. Using the rewriting rules, a term of the form $q_0(s)$, where s is an input tree over Σ, can be rewritten to an output tree t over Δ. We denote this fact by $q_0(s) \Rightarrow_M^* t$. Now the tree transformation induced by M is the set $\tau_M=\{(s,t)\in T_\Sigma\times T_\Delta \mid q_0(s) \Rightarrow_M^* t\}$, where T_Σ and T_Δ denote the set of trees over Σ and Δ, respectively.

Since trees generalize strings, more or less it should be clear that top-down tree transducers generalize gsm's. Two trees $s\in T_\Sigma$ and $t\in T_\Delta$, have the same shape if the domains of s and t are the same, i.e., they differ only in the labels of their nodes. One can also find out easily that a natural generalization of the length preserving property of gsm's for top-down tree transducers is the shape preserving property. A top-down tree transducer M is shape preserving if for every input tree $s\in T_\Sigma$ and output tree $t\in\tau_M(s)$, s and t have the same shape. For instance, a top-down relabeling tree transducer (relabeling, to be short), i.e., a top-down tree transducer of which the rules have the form $q(\sigma(x_1,\ldots,x_k))\to\delta(q_1(x_1),\ldots,q_k(x_k))$, where δ is an output symbol of arity k from Δ is obviously shape preserving. Note that top-down relabeling tree transducers generalize Mealy automata.

As the main result of this paper we prove two things. Namely, we show that every shape preserving top-down tree transducer is equivalent to a top-down relabeling tree transducer. This result naturally generalizes the corresponding one for gsm's. Moreover we prove that it is decidable if a top-down tree transducer is shape preserving.

Although we cannot give a direct practical application of this generalization result, we think that it enriches the theory of tree transducers with a theorem that carries over from gsm's to top-down tree transducers.

To support this, we recall a result concerning top-down relabeling tree transformations, which, using the results of this paper, can be generalized to shape preserving top-down tree transformations.

In [6] the iteration of length preserving gsm transductions was considered and several interesting results were obtained on the *UCI* closure of length preserving gsm transductions and of length preserving functional gsm transductions, where U, C and I mean union, composition and iteration, respectively. Recently Z. Fülöp and A. Terlutte were going to generalize the results of [6] to the class of shape preserving top-down tree transducers. However, they succeeded only partially because they were able to generalize those results only to the class of relabeling tree transducers, which as mentioned are special shape preserving top-down tree transducers. In fact, in [3] they considered the closure class $UCI(QREL)$, where $QREL$ is the class of top-down relabeling tree transformations. They gave a characterization of $UCI(QREL)$ in terms of a short expression built up from $QREL$ with composition and iteration. They also gave a characterization of $UCI(QREL)$ in terms of one-step rewrite relations of very simple term rewrite systems. They gave a similar characterization of $UC(QFREL_+)$ where $QFREL_+$ is the class consisting of the transitive closures of all functional relabeling tree transformations. Finally they showed that $UCI(QREL) = UCI(QFREL)$. Now, in the light of the equivalence of shape preserving top-down tree transducers and top-down relabeling tree transducers (Theorem 3.30), $QREL$ and $QFREL$ in the above description can be replaced by the class of the shape preserving top-down tree transformations and functional shape preserving top-down tree transformations, i.e., we can say that the results of [6] can be generalized to (unrestricted) shape preserving top-down tree transducers, respectively.

As regards the organization of the paper, the necessary definitions and the terminology are introduced in Section 2. The main results discussed above are in Section 3. In Section 4 we present some conclusions and further research problems.

2. Definitions and preliminary results

The set of nonnegative integers is denoted by $\mathbb{N}$. For $k \in \mathbb{N}$, $[k]$ denotes the set $\{1, \ldots, k\}$. The cardinality of a set A is denoted by $\|A\|$.

An *alphabet* A is a finite, nonempty set of symbols. We denote by A^* the set of *strings* (or *words*) *over* A, we let $A^+ = A - \{\varepsilon\}$, where ε is the *empty string*. For a string $w \in A^*$ and an integer $k \geqslant 0$, we denote by w^k the string $ww \ldots w$, where w appears k times. A string $u \in A^*$ is the *prefix* of a $w \in A^*$ if there is a $v \in A^*$ such that $uv = w$. Moreover u and w are *incomparable* if neither u is a prefix of w nor w is a prefix of u. The *length* of a string $w \in A^*$ is defined in the usual way and is denoted by $length(w)$. Moreover, for every $k \in \mathbb{N}$, we put $A^{*,k} = \{w \in A^* \mid length(w) \leqslant k\}$. The ith letter of a string w is denoted by $w(i)$.

A *ranked alphabet* is a pair $(\Sigma, rank)$, where Σ is an alphabet and $rank$ is a mapping from Σ to $\mathbb{N}$. For every $k \geqslant 0$, we denote by $\Sigma^{(k)}$ the set of symbols $\sigma \in \Sigma$ with $rank(\sigma) = k$ and, for a symbol $\sigma \in \Sigma$ we write $\sigma^{(k)}$ to denote that $\sigma \in \Sigma^{(k)}$.

Let A be a set disjoint with Σ. The set of (*finite*, *labeled and ordered*) *trees over* Σ *indexed by* A, denoted by $T_\Sigma(A)$, is the smallest subset T of $(\Sigma \cup A \cup \{(,)\} \cup \{,\})^*$, such that (i) $A \subseteq T$ and (ii) if $\sigma \in \Sigma^{(k)}$ with $k \geqslant 0$ and $s_1, \ldots, s_k \in T$, then $\sigma(s_1, \ldots, s_k) \in T$. In case $k = 0$, we identify $\sigma(\,)$ with σ. Moreover, $T_\Sigma(\emptyset)$ is denoted by T_Σ. It should be clear that $T_\Sigma = \emptyset$ if and only if $\Sigma^{(0)} = \emptyset$. Since we are not interested in this particular case, we assume that $\Sigma^{(0)} \neq \emptyset$ for every ranked alphabet Σ appearing as input or output ranked alphabet of some tree transducer in this paper.

A *tree language* is a subset of T_Σ while a *tree transformation* is a subset of $T_\Sigma \times T_\Delta$, where Σ and Δ are ranked alphabets. For a tree transformation $\tau \subseteq T_\Sigma \times T_\Delta$, we denote the *domain* and the *range* of τ by $dom(\tau)$ and $ran(\tau)$, respectively.

We will need the set $X = \{x_1, x_2, \ldots\}$ of *variable symbols*. For every $k \geqslant 0$, we define $X_k = \{x_1, \ldots, x_k\}$, thus $X_0 = \emptyset$. We use the variables to occur in trees, so we will frequently consider the sets $T_\Sigma(X)$, $T_\Sigma(X_k)$, etc. of trees where Σ is a ranked alphabet. We identify $T_{\Sigma^{(1)}}(X_1)$ with $(\Sigma^{(1)})^*$.

We distinguish a subset $\widehat{T_\Sigma}(X_k)$ of $T_\Sigma(X_k)$ as follows. A tree $t \in T_\Sigma(X_k)$ is in $\widehat{T_\Sigma}(X_k)$ if for every $1 \leqslant i \leqslant k$, the variable x_i occurs exactly once in t and, reading the leaves of t from left to right, the variables occur in the order $x_1, x_2, \ldots, x_k$. Note that $\widehat{T_{\Sigma^{(1)}}}(X_1) = T_{\Sigma^{(1)}}(X_1) (= (\Sigma^{(1)})^*)$.

The *tree substitution* is defined as follows. Let $t \in T_\Sigma(X_k)$ and let $t_1, \ldots, t_k$ be also trees over (maybe other) ranked alphabets. Then $t[t_1, \ldots, t_k]$ stands for the tree which is obtained from t by substituting, for every $1 \leqslant i \leqslant k$, the tree t_i for every occurrence of x_i. If $\gamma \in (\Sigma^{(1)})^*$, then $\gamma[t]$ is also denoted by γt in order to avoid too many parentheses. Moreover, for a tree language L by γL we mean the set $\{\gamma t \mid t \in L\}$.

If Q is a unary ranked alphabet, i.e., the rank of all symbols in Q is 1, and Y is a finite subset of X, then $Q(Y)$ stands for the set $\{q(x_i) \mid q \in Q$ and $x_i \in Y\}$.

Now we introduce some characteristics of trees, namely we define the height and the set of occurrences of a tree.

Let Σ be a ranked alphabet and A be a set. For an arbitrary $s \in T_\Sigma(A)$ the *height* of s ($height(s)$) and the set of *occurrences* of s ($occ(s)$) is defined as follows

(i) If $s \in \Sigma^{(0)} \cup A$, then $height(s) = 1$, $occ(s) = \{\varepsilon\}$.
(ii) If $s = \sigma(s_1, \ldots, s_k)$ for some $\sigma \in \Sigma^{(k)}$, $k \geqslant 1$ and $s_1, \ldots, s_k \in T_\Sigma(A)$, then
- $height(s) = 1 + max\{height(s_i) \mid 1 \leqslant i \leqslant k\}$,
- $occ(s) = \{\varepsilon\} \cup \{w \mid w = iv, 1 \leqslant i \leqslant k, v \in occ(s_i)\}$.

Obviously, $height(s) \in \mathbb{N}$, while $occ(s) \subseteq \mathbb{N}^*$.

Also, for $s \in T_\Sigma(A)$, and $w \in occ(s)$, we define the *subtree* at w of s ($stree(s, w)$) as follows.

(i) If $s \in \Sigma^{(0)} \cup A$ (and thus $w = \varepsilon$), then $stree(s, w) = s$.
(ii) If $s = \sigma(s_1, \ldots, s_k)$ for some $\sigma \in \Sigma^{(k)}$ with $k \geqslant 1$ and $s_1, \ldots, s_k \in T_\Sigma(A)$, then
- if $w = \varepsilon$, then $stree(s, w) = s$, otherwise,
- if $w = iv$ for some $1 \leqslant i \leqslant k$, then $stree(s, w) = stree(s_i, v)$.

Hence $stree(s, w) \in T_\Sigma(A)$.

For trees $s, t \in T_\Sigma$, *t is a subtree of s* if there is a $w \in occ(s)$ with $stree(s, w) = t$.

Let $s \in \widehat{T_\Sigma}(X_1)$. We denote by $occ(s, x_1)$ the unique occurrence $w \in occ(s)$ for which $stree(s, w) = x_1$.

Let Σ and Δ be a ranked alphabets. Two trees $s \in T_\Sigma$ and $t \in T_\Delta$ "*have the same shape*", denoted by $s \approx t$, if $occ(s) = occ(t)$. If $\Delta = \Sigma$, then $\approx$ is an equivalence relation over T_Σ.

For instance, if $\Sigma = \{\sigma^{(2)}, \delta^{(2)}, \gamma^{(1)}, a^{(0)}, b^{(0)}\}$ and $\Delta = \Sigma$, then $a \approx b$, $\sigma(a, \gamma(b)) \approx \delta(b, \gamma(a))$ and $\sigma(a, \delta(b, \gamma(b))) \approx \delta(b, \sigma(b, \gamma(a)))$.

If s and t do not have the same shape, then we write $s \not\approx t$.

A tree transformation $\tau \subseteq T_\Sigma \times T_\Delta$ is *shape preserving* if, for every $(s, t) \in \tau$, $s \approx t$. A tree language $L \subseteq T_\Sigma$ is called *uniform* if, for every $s, t \in L$, we have $s \approx t$. Note that a uniform tree language is finite.

Next we introduce the concept of a *tree homomorphism*. Let Σ and Δ be ranked alphabets and let $\bar{h} : \Sigma \to T_\Delta(X)$ be a mapping with the property that if $\sigma \in \Sigma_k$ for some $k \geqslant 0$, then $\bar{h}(\sigma) \in T_\Delta(X_k)$ holds. The tree homomorphism induced by $\bar{h}$ is the mapping

$h: T_\Sigma \to T_\Delta$ defined by induction as follows:
(i) If $\sigma \in \Sigma_0$, then $h(\sigma) = \bar{h}(\sigma)$.
(ii) If $\sigma \in \Sigma_k$ for some $k \geqslant 1$ and $s_1, \ldots, s_k \in T_\Sigma$, then $h(\sigma(s_1, \ldots, s_k)) = \bar{h}(\sigma)[h(s_1), \ldots, h(s_k)]$.

A *top-down tree transducer* is a system $M = (Q, \Sigma, \Delta, q_0, R)$, where Q is a unary ranked alphabet, called the set of *states*; Σ and Δ are ranked alphabets called the *input* and the *output ranked alphabet*, respectively, satisfying that $Q \cap (\Sigma \cup \Delta) = \emptyset$; $q_0 \in Q$ is the *initial state*; and R is a finite set of *rewriting rules* of the form $q(\sigma(x_1, \ldots, x_k)) \to r$ with $k \geqslant 0$, $\sigma \in \Sigma_k$, $q \in Q$ and $r \in T_\Delta(Q(X_k))$. Here $q(\sigma(x_1, \ldots, x_k))$ and r are called the *left-hand side* and the *right-hand side* of that rule, respectively.

The *derivation relation* induced by M is a binary relation $\Rightarrow_M$ over the set $T_{Q \cup \Sigma \cup \Delta}$ defined as follows: for $s, t \in T_{Q \cup \Sigma \cup \Delta}$, we write $s \Rightarrow_M t$ if and only if there is a rule $q(\sigma(x_1, \ldots, x_k)) \to r$ in R and t is obtained from s by replacing an occurrence of a subtree $q(\sigma(s_1, \ldots, s_k))$ of s by $r[s_1, \ldots, s_k]$, where $s_1, \ldots, s_k \in T_\Sigma$. The reflexive, transitive closure of $\Rightarrow_M$ is denoted by $\Rightarrow_M^*$. Then the tree transformation induced by M in a state $q \in Q$ is the relation

$$\tau_{M,q} = \{(s,t) \in T_\Sigma \times T_\Delta \mid q(s) \Rightarrow_M^* t\},$$

and the tree transformation induced by M is $\tau_M = \tau_{M,q_0}$.

A rule $q(\sigma(x_1, \ldots, x_k)) \to r$ in R is *useful* if it takes part in a successful derivation. More exactly, if there are $u \in \widehat{T_\Sigma}(X_1)$, $s_1, \ldots, s_k \in T_\Sigma$, $v \in \widehat{T_\Delta}(X_1)$ and $t \in T_\Delta$ such that

$$q_0(u[\sigma(s_1, \ldots, s_k)]) \Rightarrow_M^* v[q(\sigma(s_1, \ldots, s_k))] \Rightarrow_M v[r[s_1, \ldots, s_k]] \Rightarrow_M^* t.$$

It is an exercise to show that useless rules can be eliminated from a top-down tree transducer. A state $q \in Q$ is useful if it is on the left-hand side of a useful rule. Throughout the paper all tree transducers which we consider are assumed to have only useful rules and states.

A tree $s \in T_\Sigma$ is called an *input tree* to M or just an input tree. A tree $t \in T_\Delta$ satisfying $q(s) \Rightarrow_M^* t$ for some $s \in T_\Sigma$ and $q \in Q$ is called an *output tree*. Hence input trees and output trees are trees over the input and the output ranked alphabet, respectively.

A tree transformation τ is a *top-down tree transformation* if there is a top-down tree transducer M which induces τ, i.e., for which $\tau = \tau_M$ holds.

Next, we define some restrictions on top-down tree transducers. For this, let $M = (Q, \Sigma, \Delta, q_0, R)$ be a top-down tree transducer. We say that M is
(a) *linear* (*nondeleting*) if for each rule $q(\sigma(x_1, \ldots, x_k)) \to r$ in R, each of the variables $x_1, \ldots, x_k$ appears at most once (at least once) in r;
(b) a *top-down relabeling tree transducer* (or just a relabeling) if each rule in R has the form $q(\sigma(x_1, \ldots, x_k)) \to \delta(q_1(x_1), \ldots, q_k(x_k))$, where $\delta \in \Delta^{(k)}$ and $q_1, \ldots, q_k \in Q$;
(c) *shape preserving* if τ_M is shape preserving.

Two top-down tree transducers M and M' are *equivalent* if $\tau_M = \tau_{M'}$.

We introduce top-down tree automata as special relabelings because this will be convenient in what follows.

A *top-down tree automaton* is a relabeling $T = (Q, \Sigma, \Delta, q_0, R)$ such that $\Sigma = \Delta$ and each rule in R has the form $q(\sigma(x_1, \ldots, x_k)) \to \sigma(q_1(x_1), \ldots, q_k(x_k))$. Since the input and the output ranked alphabets are the same we can also write $T = (Q, \Sigma, q_0, R)$.

The tree transformation τ_T is a partial identity mapping over T_Σ. The tree language recognized by T is the domain (and hence the range) of τ_T. A tree language L is *recognizable* if there is a top-down tree automaton which recognizes L.

3. The main results

It is obvious that relabelings are shape preserving. In this section we show that, in fact, the converse also holds in the sense that every shape preserving top-down tree transducer is equivalent to a relabeling. As a byproduct, we will obtain that it is decidable if a top-down tree transducer is shape-preserving.

The proof can be divided into three parts. In the first part we show that every shape preserving top-down tree transducer is a permutation top-down quasirelabeling. A permutation top-down quasirelabeling differs from a relabeling in that the right-hand sides of its rules may contain some extra unary output symbols and a permutation of the variables is also allowed in the right-hand sides (exact definition is given below).

In the second part we show that every shape preserving permutation top-down quasirelabeling is equivalent to a top-down quasirelabeling. A top-down quasirelabeling is like a permutation top-down quasirelabeling, however only the trivial (identity) permutation of the variables is allowed in the right-hand sides of the rules.

In the third part we show that every shape preserving top-down quasirelabeling is equivalent to a relabeling.

Finally we will show that it is decidable whether a top-down tree transducer is shape preserving.

We begin the first part of our program with the definition of the permutation top-down quasirelabeling.

Definition 3.1. A top-down tree transducer $M=(Q,\Sigma,\Delta,q_0,R)$ is called a *permutation top-down quasirelabeling* if each rule in R has either of the following forms:
1. $q(\sigma(x_1,\dots,x_k))\to\gamma\delta(\gamma_1 q_1(x_{\pi(1)}),\dots,\gamma_k q_k(x_{\pi(k)}))$, where $k\neq 1$, $q,q_1,\dots,q_k\in Q$, $\sigma\in\Sigma^{(k)}$, $\delta\in\Delta^{(k)}$, $\gamma,\gamma_1,\dots,\gamma_k\in(\Delta^{(1)})^*$ and $\pi:[k]\to[k]$ is a permutation, and
2. $q(\sigma(x_1))\to\gamma p(x_1)$, where $q,p\in Q$, $\sigma\in\Sigma^{(1)}$ and $\gamma\in(\Delta^{(1)})^*$.

Notice, in case $k=0$ rules of type 1. have the form $q(\sigma)\to\gamma\delta$, where $q\in Q$, $\sigma\in\Sigma^{(0)}$, $\gamma\in(\Delta^{(1)})^*$ and $\delta\in\Delta^{(0)}$. Moreover rules of type 2. with $\gamma=\varepsilon$ have the form $q(\sigma(x_1))\to p(x_1)$.

We continue with proving that every shape preserving top-down tree transducer is nondeleting.

Lemma 3.2. *Let $M=(Q,\Sigma,\Delta,q_0,R)$ be a shape preserving top-down tree transducer. Then M is nondeleting.*

Proof. We prove by contradiction. Let us assume that there is a rule $q(\sigma(x_1,\dots,x_k))\to r$ in R such that $k\geqslant 1$ and, say, x_i does not occur in r. This rule is also useful, so there are $u \in \widehat{T_\Sigma}(X_1)$, $s_1, \ldots, s_k \in T_\Sigma$, $v \in \widehat{T_\Delta}(X_1)$ and $t \in T_\Delta$ such that

$$q_0(u[\sigma(s_1,\ldots,s_k)]) \Rightarrow_M^* v[q(\sigma(s_1,\ldots,s_k))] \Rightarrow_M v[r[s_1,\ldots,s_k]] \Rightarrow_M^* t.$$

Since M is shape preserving, $u[\sigma(s_1,\ldots,s_k)] \approx t$ holds. Now change the involved occurrence of s_i to a s_i' such that $s_i \not\approx s_i'$. Then certainly $u[\sigma(s_1,\ldots,s_k)] \not\approx u[\sigma(s_1,\ldots,s_i',\ldots,s_k)]$. On the other hand, since x_i does not occur in r, we have $r[s_1,\ldots,s_i',\ldots,s_k] = r[s_1,\ldots,s_k]$ and thus

$$q_0(u[\sigma(s_1,\ldots,s_i',\ldots,s_k)]) \Rightarrow_M^* v[q(\sigma(s_1,\ldots,s_i',\ldots,s_k))] \Rightarrow_M$$

$$v[r[s_1,\ldots,s_i',\ldots,s_k]] = v[r[s_1,\ldots,s_k]] \Rightarrow_M^* t.$$

Since M is shape preserving, $u[\sigma(s_1,\ldots,s_i',\ldots,s_k)] \approx t$, a contradiction. □

Next we define the branch number of a tree.

Definition 3.3. Let Σ be a ranked alphabet. A symbol $\sigma \in \Sigma$ is called a *branch symbol* provided its rank is greater than 1. The *branch number* $\mathrm{bn}(s)$ of a tree $s \in T_\Sigma(X)$ is defined by induction as follows.
(i) If $s \in \Sigma^{(0)}$ or $s \in X$, then $\mathrm{bn}(s) = 0$.
(ii) If $s = \sigma(s_1,\ldots,s_k)$ for some $\sigma \in \Sigma^{(k)}, k \geqslant 1$ and $s_1,\ldots,s_k \in T_\Sigma$, then
- if $k = 1$, then $\mathrm{bn}(s) = \mathrm{bn}(s_1)$,
- if $k > 1$, then $\mathrm{bn}(s) = 1 + \sum_{i=1}^{k} \mathrm{bn}(s_i)$.

Hence the branch number of a tree s is the sum of number of the occurrences of the branch symbols in s. Certainly, if $s \approx t$, then $\mathrm{bn}(s) = \mathrm{bn}(t)$.

Lemma 3.4. *Let $M = (Q, \Sigma, \Delta, q_0, R)$ be a shape preserving top-down tree transducer. Then, for every rule $q(\sigma(x_1,\ldots,x_k)) \to r$ in R, we have $\mathrm{bn}(r) \leqslant 1$.*

Proof. We prove by contradiction. Assume, there is a rule $q(\sigma(x_1,\ldots,x_k)) \to r \in R$ with $\mathrm{bn}(r) > 1$. This rule can be applied in a successful derivation $q_0(s) \Rightarrow_M^* t$ for some $s \in T_\Sigma$ and $t \in T_\Delta$. Since M is shape preserving, $\mathrm{bn}(s) = \mathrm{bn}(t)$. The application of the above rule increases the branch number of the output with respect to the input, hence another rule is needed to compensate the increase. The only chance to decrease the branch number is to apply a rule of the form $p(\delta(x_1,\ldots,x_l)) \to r$, where $l \geqslant 1$ and r does not contain some of the variables $x_1,\ldots,x_l$. However, by Lemma 3.2, there are no such rules in R. Thus $\mathrm{bn}(s) > \mathrm{bn}(t)$, which is a contradiction. Hence our statement follows. □

Corollary 3.5. *Let $M = (Q, \Sigma, \Delta, q_0, R)$ be a shape preserving top-down tree transducer. Then every rule in R has either of the following forms.*
1. $q(\sigma(x_1,\ldots,x_k)) \to \gamma\delta(\gamma_1 q_1(x_{i_1}),\ldots,\gamma_m q_m(x_{i_m}))$, *where* $k \neq 1$, $m \geqslant k$, $q, q_1,\ldots,q_m \in Q$, $\sigma \in \Sigma^{(k)}$, $\delta \in \Delta^{(m)}$, $\gamma, \gamma_1,\ldots,\gamma_m \in (\Delta^{(1)})^*$ *and* $\{x_{i_1},\ldots,x_{i_m}\} = X_k$. (*Notice, some x_j may occur more than once in the right-hand side.*)
2. $q(\sigma(x_1)) \to \gamma p(x_1)$, *where* $q, p \in Q$, $\sigma \in \Sigma^{(1)}$ *and* $\gamma \in (\Delta^{(1)})^*$.

Proof. It immediately follows from Lemmas 3.2 and 3.4. □

Next, we are going to show that in 1. of the above corollary even $k=m$ holds, which means that a shape preserving top-down tree transducer is a permutation top-down quasirelabeling. For this, we define the weighted branch number of a tree.

Definition 3.6. Let Σ be a ranked alphabet. The *weighted branch number* $\mathrm{wbn}(s)$ of a tree $s \in T_\Sigma(X)$ is defined by induction as follows.
(i) If $s \in \Sigma^{(0)}$ or $s \in X$, then $\mathrm{wbn}(s)=0$,
(ii) if $s=\sigma(s_1,\ldots,s_k)$ for some $\sigma \in \Sigma^{(k)}, k \geqslant 1$ and $s_1,\ldots,s_k \in T_\Sigma$, then
- if $k=1$, then $\mathrm{wbn}(s)=\mathrm{wbn}(s_1)$,
- if $k>1$, then $\mathrm{wbn}(s)=k+\sum_{i=1}^{k}\mathrm{wbn}(s_i)$.

Certainly, if $s \approx t$, then $\mathrm{wbn}(s)=\mathrm{wbn}(t)$.

Lemma 3.7. *Let $M=(Q,\Sigma,\Delta,q_0,R)$ be a shape preserving top-down tree transducer. Then M is a permutation top-down quasirelabeling.*

Proof. Each rule in R is as in 1. or 2. in Corollary 3.5. It is enough to prove that in case 1. only $m=k$ is possible. This can be shown easily in the following way. If $m>k$, then the application of that rule increases the weighted branch number, which increase cannot be compensated somewhere else, cf. the proof of Lemma 3.4. □

Next we give an example of a shape preserving top-down tree transducer, which, by Lemma 3.7 is a permutation top-down quasirelabeling. In this example we demonstrate that a real (not the identity) permutation of the variables in the right-hand sides of rules may occur.

Example 3.8. Let $M=(Q,\Sigma,\Delta,q_0,R)$ be a top-down tree transducer, where
- $Q=\{q_0,q_\beta,q_\gamma,q_1,q_2\}$, $\Sigma=\{\sigma^{(2)},\gamma^{(1)},\beta^{(1)},\alpha^{(0)}\}$, $\Delta=\{\delta^{(2)},\gamma^{(1)},\beta^{(1)},\alpha^{(0)}\}$,
- R is the set of the rules
  - $q_0(\sigma(x_1,x_2))\to\delta(q_\gamma(x_2),q_\beta(x_1))$,
  - $q_\gamma(\gamma(x_1))\to\gamma(q_1(x_1))$, $q_1(\gamma(x_1))\to q_2(x_1)$, $q_2(\alpha)\to\alpha$,
  - $q_\beta(\beta(x_1))\to\beta(\beta(q_2(x_1)))$.

The only successful derivation of M is

$$q_0(\sigma(\beta(\alpha),\gamma(\gamma(\alpha)))) \Rightarrow_M \delta(q_\gamma(\gamma(\gamma(\alpha))),q_\beta(\beta(\alpha))) \Rightarrow_M^* \delta(\gamma(\alpha),\beta(\beta(\alpha))),$$

hence M is shape preserving. On the other hand, the permutation of the variables in the right-hand side of the rule $q_0(\sigma(x_1,x_2))\to\delta(q_\gamma(x_2),q_\beta(x_1))$ is not the identity.

Now we begin to elaborate the second part. We start with the definition of the concept of a top-down quasirelabeling.

Definition 3.9. A top-down tree transducer $M=(Q,\Sigma,\Delta,q_0,R)$ is a *top-down quasi-relabeling* if it is a permutation top-down quasirelabeling and for every rule

$q(\sigma(x_1,\ldots,x_k))\to\gamma\delta(\gamma_1 q_1(x_{\pi(1)}),\ldots,\gamma_k q_k(x_{\pi(k)}))$ in R, π is the identity permutation, i.e., $\pi(1)=1,\ldots,\pi(k)=k$.

In what follows we develop a procedure which transforms a shape preserving permutation top-down quasirelabeling M into an equivalent top-down quasirelabeling M'. For this, however, we need the following preparation.

Definition 3.10. Let $M=(Q,\Sigma,\Delta,q_0,R)$ be a permutation top-down quasirelabeling and let

$$\mu : q(\sigma(x_1,\ldots,x_k))\to\gamma\delta(\gamma_1 q_1(x_{\pi(1)}),\ldots,\gamma_k q_k(x_{\pi(k)}))$$

be a rule in R, where $k>1$, $q,q_1,\ldots,q_k\in Q$, $\sigma\in\Sigma^{(k)}$, $\delta\in\Delta^{(k)}$, $\gamma,\gamma_1,\ldots,\gamma_k\in(\Delta^{(1)})^*$ and $\pi : [k]\to[k]$ is a permutation. The *permutation degree* of μ is the number of indexes $1\leqslant i\leqslant k$ for which $\pi(i)\neq i$. A rule with permutation degree greater than one is called a *permutation rule*. Moreover, a state $q\in Q$ is a *permutation state* if there is a permutation rule of the above form. The permutation degree of M is the sum of the permutation degrees of its rules of the above form.

Notice that the permutation degree of a top-down quasirelabeling and thus of a relabeling is 0.

Lemma 3.11. *Let $M=(Q,\Sigma,\Delta,q_0,R)$ be a shape preserving permutation top-down quasirelabeling. Then for every $k>1$, permutation rule $q(\sigma(x_1,\ldots,x_k))\to\gamma\delta(\gamma_1 q_1(x_{\pi(1)})$ $,\ldots,\gamma_k q_k(x_{\pi(k)}))$ in R, where $q,q_1,\ldots,q_k\in Q$, $\sigma\in\Sigma^{(k)}$, $\delta\in\Delta^{(k)}$, $\gamma,\gamma_1,\ldots,\gamma_k\in(\Delta^{(1)})^*$ and $\pi : [k]\to[k]$ is a permutation, and $1\leqslant i\leqslant k$, if $\pi(i)\neq i$, then both $dom(\tau_{M,q_i})$ and $ran(\tau_{M,q_i})$ are uniform.*

Proof. Let $q(\sigma(x_1,\ldots,x_k))\to\gamma\delta(\gamma_1 q_1(x_{\pi(1)}),\ldots,\gamma_k q_k(x_{\pi(k)}))$ be a permutation rule in R and assume that for some $1\leqslant i\leqslant k$, $\pi(i)\neq i$ holds.

Since the rule is useful, there are $u\in\widehat{T_\Sigma}(X_1)$, $s,s_1,\ldots,s_k\in T_\Sigma$, $v\in\widehat{T_\Delta}(X_1)$ and $t,t_1,\ldots,t_k\in T_\Delta$ such that

$$\begin{aligned} q_0(s) &= q_0(u[\sigma(s_1,\ldots,s_k)]) \\ &\Rightarrow_M^* v[q(\sigma(s_1,\ldots,s_k))] \\ &\Rightarrow_M v[\gamma\delta(\gamma_1 q_1(s_{\pi(1)}),\ldots,\gamma_k q_k(s_{\pi(k)}))] \qquad (\dagger) \\ &\Rightarrow_M^* v[\gamma\delta(\gamma_1 t_1,\ldots,\gamma_k t_k)] \\ &= t. \end{aligned}$$

Hence, for every $1\leqslant j\leqslant k$, $q_j(s_{\pi(j)})\Rightarrow_M^* t_j$. Moreover, $s\approx t$. Now we distinguish three cases.

Case 1: $occ(u,x_1)$ and $occ(v,x_1)$ are incomparable, see Fig. 1. Let us suppose that $dom(\tau_{M,q_i})$ is not uniform. Since $stree(s,occ(u,x_1)\pi(i))=s_{\pi(i)}$ and since M is shape preserving, we have $stree(t,occ(u,x_1)\pi(i))\approx s_{\pi(i)}$. Now change that occurrence

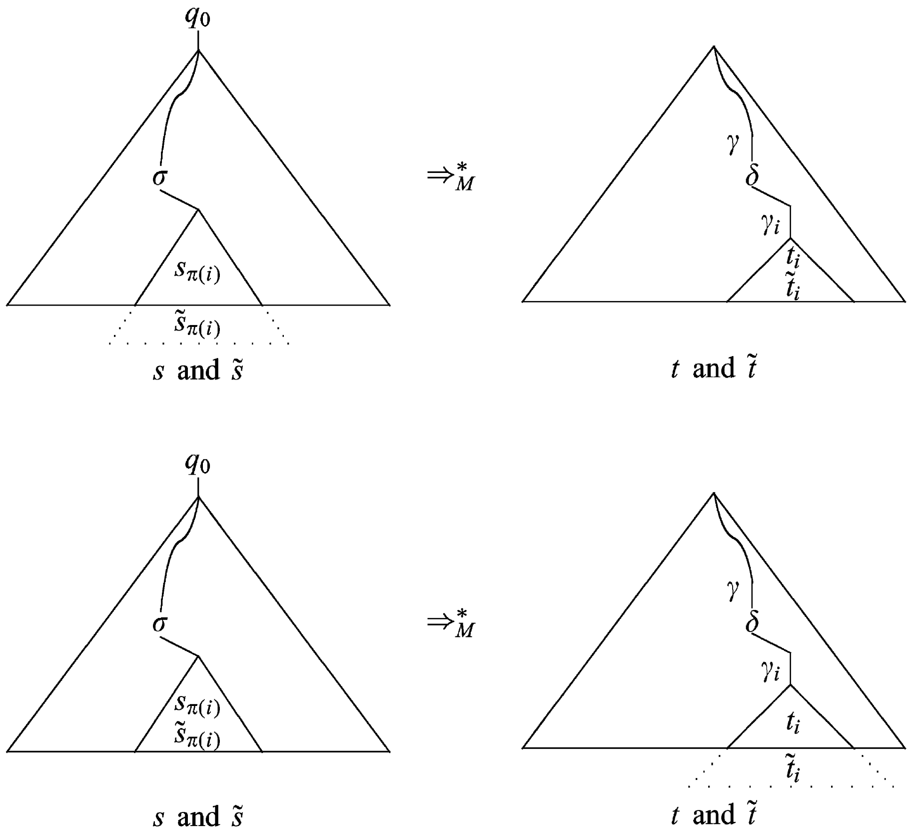

Fig. 1. The sets $dom(\tau_{M,q_i})$ and $ran(\tau_{M,q_i})$ are uniform, Case 1.

of $s_{\pi(i)}$ in s to a $\tilde{s}_{\pi(i)} \in dom(\tau_{M,q_i})$ such that $s_{\pi(i)} \not\approx \tilde{s}_{\pi(i)}$ and form the input tree $\tilde{s} = u[\sigma(\ldots, \tilde{s}_{\pi(i)}, \ldots)]$. (Such a $\tilde{s}_{\pi(i)}$ exists because $dom(\tau_{M,q_i})$ is not uniform.)

Since $\tilde{s}_{\pi(i)} \in dom(\tau_{M,q_i})$, there is a tree $\tilde{t} \in T_\Delta$ such that $q_0(\tilde{s}) \Rightarrow_M^* \tilde{t}$ and since M is shape preserving, $\tilde{s} \approx \tilde{t}$. Now $stree(\tilde{s}, occ(u,x_1)\pi(i)) = \tilde{s}_{\pi(i)}$ and since $occ(u,x_1)$ and $occ(v,x_1)$ are incomparable

$$stree(\tilde{t}, occ(u,x_1)\pi(i)) = stree(t, occ(u,x_1)\pi(i)) \approx s_{\pi(i)},$$

which contradicts $\tilde{s} \approx \tilde{t}$.

Now assume that $ran(\tau_{M,q_i})$ is not uniform. Let $n = length(\gamma)$. Since $stree(t, occ(v, x_1)1^n i) = \gamma_i t_i$ and since M is shape preserving, we have $stree(s, occ(v,x_1)1^n i) \approx \gamma_i t_i$. Now change the involved occurrence of $s_{\pi(i)}$ in s to a $\tilde{s}_{\pi(i)} \in dom(\tau_{M,q_i})$ such that $q_i(\tilde{s}_{\pi(i)}) \Rightarrow_M^* \tilde{t}_i$ and $t_i \not\approx \tilde{t}_i$, and denote the resulting tree by $\tilde{s}$. (Such a $\tilde{s}_{\pi(i)}$ exists because $ran(\tau_{M,q_i})$ is not uniform.)

Since $\tilde{s}_{\pi(i)} \in dom(\tau_{M,q_i})$, there is a tree $\tilde{t} \in T_\Delta$ such that $q_0(\tilde{s}) \Rightarrow_M^* \tilde{t}$ and since M is shape preserving, $\tilde{s} \approx \tilde{t}$. Now $stree(\tilde{t}, occ(v,x_1)1^n i) = \gamma_i \tilde{t}_i$ and since $occ(u,x_1)$ and

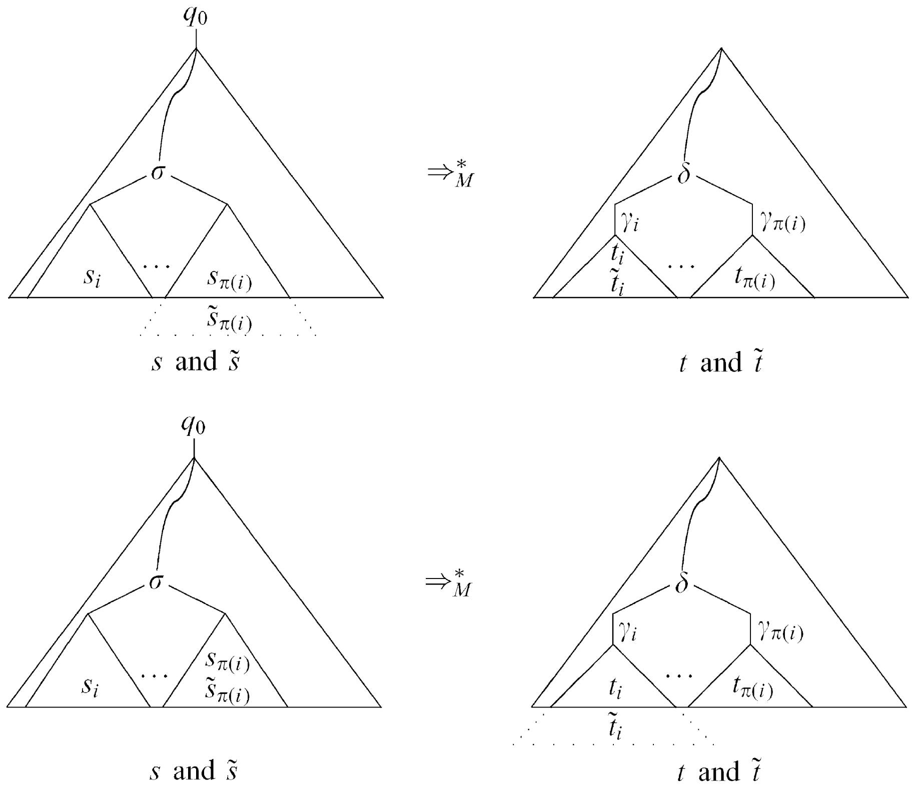

Fig. 2. The sets $dom(\tau_{M,q_i})$ and $ran(\tau_{M,q_i})$ are uniform, Case 2.

$occ(v,x_1)$ are incomparable

$$stree(\tilde{s}, occ(v,x_1)1^n i) = stree(s, occ(v,x_1)1^n i) \approx \gamma_i t_i,$$

which contradicts $\tilde{s} \approx \tilde{t}$.

Notice that in this case we need not use the assumption $\pi(i) \neq i$.

Case 2: $occ(v,x_1)$ is a prefix of $occ(u,x_1)$, see Fig. 2. Now $occ(u,x_1) = occ(v[\gamma x_1],x_1)$ because M is a shape preserving permutation top-down quasirelabeling and thus the symbol δ written by the application of the involved rule

$$q(\sigma(x_1,\ldots,x_k)) \to \gamma\delta(\gamma_1 q_1(x_{\pi(1)}),\ldots,\gamma_k q_k(x_{\pi(k)}))$$

matches the σ being at $occ(u,x_1)$ in s. Thus, for every $1 \leqslant j \leqslant k$, $s_j \approx \gamma_j(t_j)$.

Now assume $dom(\tau_{M,q_i})$ is not uniform. (Note that $s_{\pi(i)} \in dom(\tau_{M,q_i})$.) Now change $s_{\pi(i)}$ to a $\tilde{s}_{\pi(i)} \in dom(\tau_{M,q_i})$ such that $s_{\pi(i)} \not\approx \tilde{s}_{\pi(i)}$ and form the input tree $\tilde{s} = u[\sigma(\ldots, \tilde{s}_{\pi(i)},\ldots)]$. Consider the following derivation, where in the arguments of σ and δ we indicated the $\pi(i)$th position.

$$\begin{aligned} q_0(\tilde{s}) &= q_0(u[\sigma(\ldots,\tilde{s}_{\pi(i)},\ldots)]) \\ &\Rightarrow_M^* v[q(\sigma(\ldots,\tilde{s}_{\pi(i)},\ldots))] \\ &\Rightarrow_M v[\gamma\delta(\ldots,\gamma_{\pi(i)}q_{\pi(i)}(s_{\pi^2(i)}),\ldots)] \\ &\Rightarrow_M^* v[\gamma\delta(\ldots,\gamma_{\pi(i)}t_{\pi(i)},\ldots)] \\ &= \tilde{t}. \end{aligned}$$

Since $\pi(i)\neq i$, the tree $s_{\pi^2(i)}$ remains unchanged (even if $\pi^2(i)=i$) and thus, since M is shape preserving, $\tilde{s}_{\pi(i)}\approx\gamma_{\pi(i)}t_{\pi(i)}$ must hold. This is a contradiction, because by the above observation, $s_{\pi(i)}\approx\gamma_{\pi(i)}t_{\pi(i)}$.

Now assume $ran(\tau_{M,q_i})$ is not uniform. Now change $s_{\pi(i)}$ to a $\tilde{s}_{\pi(i)}\in dom(\tau_{M,q_i})$ such that $q_i(\tilde{s}_{\pi(i)})\Rightarrow_M^*\tilde{t}_i$ and $t_i\not\approx\tilde{t}_i$. Since $\pi(i)\neq i$, the tree s_i remains unchanged and thus, since M is shape preserving, also $s_i\approx\gamma_i\tilde{t}_i$.

Then, by $s_i\approx\gamma_i t_i$, we obtain $t_i\approx\tilde{t}_i$, which is a contradiction.

Case 3: $occ(u,x_1)$ is a proper prefix of $occ(v,x_1)$. Then $occ(u,x_1)$ is also a proper prefix of $occ(v[\gamma x_1],x_1)$. On the other hand, similarly as in Case 2, it is seen that $occ(u,x_1)=occ(v[\gamma x_1],x_1)$, which is a contradiction. □

We will need the following result.

Lemma 3.12. *Let $\rho=\{(s_1,t_1),\ldots,(s_n,t_n)\}\subseteq T_\Sigma\times T_\Delta$ be finite relation and $\gamma\in(\Delta^{(1)})^*$, where Σ and Δ are ranked alphabets such that the set $\{s_1,\gamma t_1,\ldots,s_n,\gamma t_n\}$ is uniform. Then there is a top-down quasirelabeling $M=(Q,\Sigma,\Delta,q_0,R)$ such that $\tau_M=\rho$.*

Proof. In case $\gamma=\varepsilon$ the statement is clear because we can construct M as the disjoint union of the relabelings M_j which induce the relations $\{(s_j,t_j)\}$. Hence, in this particular case M is a relabeling.

Now let us assume that $\gamma\in(\Delta^{(1)})^+$ with $length(\gamma)=m$. Then, for every $1\leqslant j\leqslant n$, there are $\gamma_j\in(\Sigma^{(1)})^+$ and $u_j\in T_\Sigma$ such that $length(\gamma_j)=m$ and $\gamma_j u_j=s_j$. Obviously $\{u_1,t_1,\ldots,u_n,t_n\}$ is uniform, so, by the discussion of the case $\gamma=\varepsilon$, there is a relabeling $M'=(Q',\Sigma,\Delta,q_0',R')$ such that $\tau_{M'}=\{(u_1,t_1),\ldots,(u_n,t_n)\}$.

Let q_0 and, for every $1\leqslant j\leqslant n$, $p_{j1},\ldots,p_{j(m-1)}$ be new states. Moreover, construct the rules

$$\begin{aligned} q_0(\gamma_j(1)(x_1)) &\to p_{j1}(x_1), \\ p_{j1}(\gamma_j(2)(x_1)) &\to p_{j2}(x_1), \\ &\ldots \\ p_{j(m-1)}(\gamma_j(m)(x_1)) &\to q_0'(x_1). \end{aligned}$$

(Recall that $\gamma_j(i)$ is the ith letter of γ_j. In case $m=1$, we have the only rule $q_0(\gamma_j(1)(x_1))\to q_0'(x_1)$.)

Now, let $M = (Q, \Sigma, \Delta, q_0, R)$, where $Q = Q' \cup \{q_0\} \cup \{p_{ji} \mid 1 \leqslant j \leqslant n, 1 \leqslant i \leqslant m-1\}$ and let R be the set of the rules constructed above and of the rules in R'. It should be clear that M is a top-down quasirelabeling and $\tau_M = \rho$. □

Next, we define a relation $>$ on the state set of a permutation top-down quasirelabeling.

Definition 3.13. Let $M = (Q, \Sigma, \Delta, q_0, R)$ be a permutation top-down quasirelabeling. We define the binary relation $>$ over Q in the following way: for every $p, q \in Q$, let $p > q$ if and only if there exist $u, u' \in \widehat{T_\Sigma}(X_1)$ and $v, v' \in \widehat{T_\Delta}(X_1)$, such that the following conditions hold:

- $q_0(u[u']) \Rightarrow_M^* v[p(u')] \Rightarrow_M^+ v[v'[q(x_1)]]$,
- $occ(u, x_1)$ and $occ(v, x_1)$ are not incomparable,
- $occ(u[u'], x_1)$ and $occ(v[v'], x_1)$ are incomparable.

Notice that u may be x_1, however u' cannot be x_1 in the above definition.

Intuitively, $p > q$ if there are $u, u' \in \widehat{T_\Sigma}(X_1)$ and $v, v' \in \widehat{T_\Delta}(X_1)$ and a derivation $q_0(u) \Rightarrow_M^* v[p(x_1)]$ such that no piece of the path α leading from the root of u to x_1 takes part in a permutation during that derivation, moreover, there is a derivation $p(u') \Rightarrow_M^+ v'[q(x_1)]$ such that a piece of the path β leading from the root of u' to x_1 takes part in a permutation during that derivation. (Note that in $q_0(u) \Rightarrow_M^* v[p(x_1)]$ a permutation rule might be applied on α, which however did not move the involved piece of α. Moreover, a permutation rule was applied in $p(u') \Rightarrow_M^+ v'[q(x_1)]$ on the path β which did move the involved piece of β.)

Example 3.14. Let $M = (Q, \Sigma, \Delta, q_0, R)$ be a top-down tree transducer, where

- $Q = \{q_0, \ldots, q_9\}$, $\Sigma = \{\sigma_1^{(1)}, \sigma_2^{(3)}, \delta_1^{(1)}, \delta_2^{(1)}, \gamma_1^{(2)}, \gamma_2^{(2)}, \alpha_1^{(0)}, \alpha_2^{(0)}, \beta_1^{(0)}, \beta_2^{(0)}\}$,
  $\Delta = \{\sigma_1'^{(1)}, \sigma_2'^{(3)}, \delta_1'^{(1)}, \delta_2'^{(1)}, \gamma_1'^{(2)}, \gamma_2'^{(2)}, \alpha_1'^{(0)}, \alpha_2'^{(0)}, \beta_1'^{(0)}, \beta_2'^{(0)}\}$,
- R is the set of the rules
  - $q_0(\sigma_1(x_1)) \to \sigma_1'(q_1(x_1))$,
  - $q_1(\sigma_2(x_1, x_2, x_3)) \to \sigma_2'(q_2(x_2), q_3(x_1), q_1(x_3))$,
  - $q_1(\sigma_2(x_1, x_2, x_3)) \to \sigma_2'(q_2(x_1), q_3(x_2), q_1(x_3))$,
  - $q_2(\delta_1(x_1)) \to \delta_1'(q_4(x_1))$,
  - $q_3(\delta_2(x_1)) \to \delta_2'(q_5(x_1))$,
  - $q_4(\gamma_1(x_1, x_2)) \to \gamma_1'(q_6(x_2), q_7(x_1))$,
  - $q_5(\gamma_2(x_1, x_2)) \to \gamma_2'(q_8(x_1), q_9(x_2))$,
  - $q_6(\beta_1) \to \beta_1'$, $q_7(\alpha_1) \to \alpha_1'$, $q_8(\alpha_2) \to \alpha_2'$, $q_9(\beta_2) \to \beta_2'$,
  - $q_1(\alpha_1) \to \alpha_1'$.

For instance $q_1 > q_5$ because with $u = \sigma_1(x_1)$, $u' = \sigma_2(\delta_2(x_1), \delta_1(\gamma_1(\alpha_1, \beta_1)), \alpha_1)$, $v = \sigma_1'(x_1)$, and $v' = \sigma_2'(\delta_1'(\gamma_1'(\beta_1', \alpha_1')), \delta_2'(x_1), \alpha_1')$ we get the derivation

$$\begin{aligned} q_0(u[u']) &= q_0(\sigma_1(\sigma_2(\delta_2(x_1), \delta_1(\gamma_1(\alpha_1, \beta_1)), \alpha_1))) \\ &\Rightarrow_M^* \sigma_1'(q_1(\sigma_2(\delta_2(x_1), \delta_1(\gamma_1(\alpha_1, \beta_1)), \alpha_1))) \\ &\Rightarrow_M^* \sigma_1'(\sigma_2'(\delta_1'(\gamma_1'(\beta_1', \alpha_1')), \delta_2'(q_5(x_1)), \alpha_1')) \\ &= v[v'[q_5(x_1)]]. \end{aligned}$$

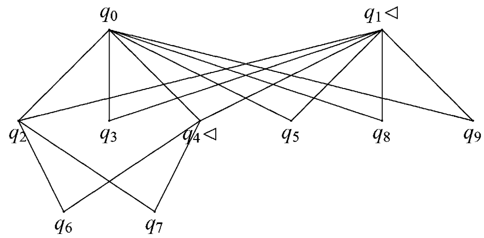

Fig. 3. The relation $>$ for M appearing in Example 3.14.

We can also see easily that $q_1 > q_4$ with $u = \sigma_1(x_1)$, $u' = \sigma_2(\delta_2(\gamma_2(\alpha_2, \beta_2)), \delta_1(x_1), \alpha_1)$, $v = \sigma_1'(x_1)$ and $v' = \sigma_2'(\delta_1'(x_1), \delta_2'(\gamma_2'(\alpha_2', \beta_2')), \alpha_1')$ and that $q_4 > q_7$ with $u = \sigma_1(\sigma_2(\delta_1(x_1), \delta_2(\gamma_2(\alpha_2, \beta_2)), \alpha_1))$, $u' = \gamma_1(x_1, \beta_1)$, $v = \sigma_1'(\sigma_2'(\delta_1'(x_1), \delta_2'(\gamma_2'(\alpha_2', \beta_2')), \alpha_1'))$ and $v' = \gamma_1'(\beta_1', x_1)$.

The full Hasse diagram of the relation $>$ on Q can be seen on Fig. 3. Note that q_1 and q_4 are permutation states, therefore we marked them by a $\lhd$.

Next we prove that the relation $>$ is computable by an algorithm for every permutation top-down quasirelabeling $M = (Q, \Sigma, \Delta, q_0, R)$.

Lemma 3.15. *For every permutation top-down quasirelabeling $M = (Q, \Sigma, \Delta, q_0, R)$, the relation $>$ is computable.*

Proof. Let $n = \|Q\|$ and $p, q \in Q$. In order to verify whether there are $u \in \widehat{T_\Sigma}(X_1)$ and $v \in \widehat{T_\Delta}(X_1)$ such that $q_0(u) \Rightarrow_M^* v[p(x_1)]$ and that no permutation happens on the path α from the root of u to the only x_1, it is sufficient to consider trees u of height at most $2n$. In fact, we may assume without loss of generality that if there is such an u, then the length of α is at most n, otherwise we can apply standard pumping arguments (cf. Lemma 10.1 in Chapter II. of [4], also Proposition 5.2 in [5])) due to the fact that M is a permutation top-down quasirelabeling. For the same reason, we can assume that the length of any path which leads from a node being in α to an arbitrary terminal node of u is at most n. Hence the height of u is at most $2n$.

In order to verify whether there are $u' \in \widehat{T_\Sigma}(X_1)$ and $v' \in \widehat{T_\Delta}(X_1)$ such that $p(u') \Rightarrow_M^* v'[q(x_1)]$ and that a permutation happens on the path β from the root of u' to the only x_1, it is sufficient to consider trees u' of height at most $3n$. In fact, we may assume without loss of generality that if there is such an u', then the length of β is at most $2n$ (n from the root of u' to the node where the permutation rule was applied and n from that node to x_1). Analogously to the previous case, we can assume that the length of any path which leads from a node being in β to an arbitrary terminal node of u' is at most n. Hence the height of u' is at most $3n$.

Thus it is decidable if $p > q$ holds. $\square$

Next we prove that the relation $>$ is a partial order, provided M is shape preserving.

Lemma 3.16. *The relation* $>$ *is a partial order for any shape preserving permutation top-down quasirelabeling* $M=(Q,\Sigma,\Delta,q_0,R)$.

Proof. We show that $>$ is irreflexive and transitive. In fact, the transitivity can be proved easily by using standard arguments, hence we leave this part of the proof.

We prove the irreflexivity by contradiction. Let us suppose there is a $p\in Q$ such that $p>p$ holds. Then there exist $u,u'\in\widehat{T_\Sigma}(X_1)$ and $v,v'\in\widehat{T_\Delta}(X_1)$ such that

$$q_0(u[u'])\Rightarrow_M^* v[p(u')]\Rightarrow_M^* v[v'[p(x_1)]]$$

and, moreover, $occ(u,x_1)$ and $occ(v,x_1)$ are not incomparable but $occ(u[u'],x_1)$ and $occ(v[v'],x_1)$ are incomparable. Note that, since $p(u')\Rightarrow^* v'[p(x_1)]$ and $u'\neq x_1$, $dom(\tau_{M,p})$ is infinite, hence not uniform. Moreover, the conditions that $occ(u,x_1)$ and $occ(v,x_1)$ are not incomparable but $occ(u[u'],x_1)$ and $occ(v[v'],x_1)$ are incomparable mean that, during the derivation $p(u')\Rightarrow_M^* v'[p(x_1)]$, a permutation rule was applied somewhere on the path leading from the root of u' to the only x_1 in it. More formally, there are $u_1'\in\widehat{T_\Sigma}(X_1)$, $\sigma\in\Sigma^{(k)}$ with $k>1$, an index $1\leqslant i\leqslant k$, further trees $s_1,\dots,s_{i-1},s_{i+1},\dots,s_k\in T_\Sigma$, $s_i\in\widehat{T_\Sigma}(X_1)$ such that $u'=u_1'[\sigma(s_1,\dots,s_k)]$. Moreover, there is a rule $q(\sigma(x_1,\dots,x_k))\to\gamma\delta(\gamma_1q_1(x_{\pi(1)}),\dots,\gamma_kq_k(x_{\pi(k)}))$, such that $\pi(i)\neq i$, and there are trees $v_1'\in\widehat{T_\Delta}(X_1)$, $t_1,\dots,t_k$ such that $t_j=t_j'[p(x_1)]$ for some $t_j'\in\widehat{T_\Delta}(X_1)$ provided $j=\pi^{-1}(i)$ and $t_j\in T_\Delta$ otherwise and the following conditions hold.

$$\begin{aligned}
p(u') = {} & p(u_1'[\sigma(s_1,\dots,s_k)])\\
\Rightarrow_M^* {} & v_1'[q(\sigma(s_1,\dots,s_k))]\\
\Rightarrow_M {} & v_1'[\gamma\delta(\gamma_1q_1(s_{\pi(1)}),\dots,\gamma_kq_k(s_{\pi(k)}))]\\
\Rightarrow_M^* {} & v_1'[\gamma\delta(\gamma_1t_1,\dots,\gamma_kt_k)]\\
= {} & v'[p(x_1)].
\end{aligned}$$

Hence $q_{\pi^{-1}(i)}(s_i)\Rightarrow_M^* t_{\pi^{-1}(i)}=t'_{\pi^{-1}(i)}[p(x_1)]$. Now, since $\pi(i)\neq i$ also $\pi^{-1}(i)\neq i$ and thus $\pi(\pi^{-1}(i))\neq\pi^{-1}(i)$. Hence, by Lemma 3.11, $dom(\tau_{M,q_{\pi^{-1}(i)}})$ is uniform. On the other hand $dom(\tau_{M,q_{\pi^{-1}(i)}})$ is infinite because $q_{\pi^{-1}(i)}(s_i)\Rightarrow_M^* t'_{\pi^{-1}(i)}[p(x_1)]$ and $dom(\tau_{M,p})$ is infinite. This means that $dom(\tau_{M,q_{\pi^{-1}(i)}})$ cannot be uniform, which is a contradiction. This proves $p\not> p$. □

Now we can show that every shape preserving permutation top-down quasirelabeling is effectively equivalent to a top-down quasirelabeling.

Lemma 3.17. *For every shape preserving permutation top-down quasirelabeling* $M=(Q,\Sigma,\Delta,q_0,R)$ *a top-down quasirelabeling* $M'=(Q',\Sigma,\Delta,q_0',R')$ *can be constructed such that* $\tau_M=\tau_{M'}$.

Proof. If the permutation degree of M is 0, then M is a top-down quasirelabeling thus we are ready.

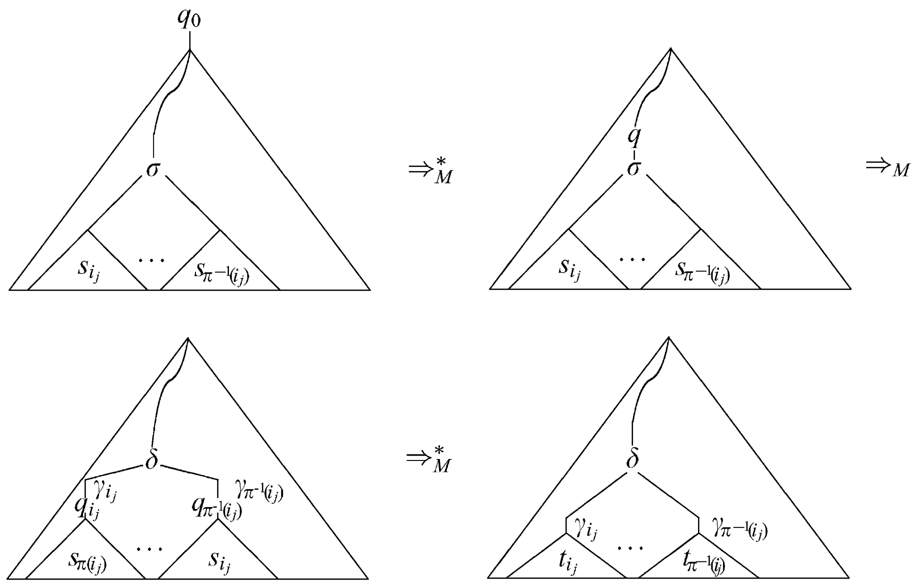

Fig. 4. $s_{i_j} \approx \gamma_{i_j} t_{i_j}$, hence the set $dom(\tau_{M,q_{\pi^{-1}(i_j)}}) \cup \gamma_{i_j} ran(\tau_{M,q_{i_j}})$ is uniform.

Otherwise, it is sufficient to show that a permutation top-down quasirelabeling $N = (\bar{Q}, \Sigma, \Delta, \bar{q_0}, \bar{R})$ can be constructed such that $\tau_M = \tau_N$ and the permutation degree of N is less than that of M.

To see this, let us take a permutation rule

$$\mu : q(\sigma(x_1,\dots,x_k)) \to \gamma\delta(\gamma_1 q_1(x_{\pi(1)}),\dots,\gamma_k q_k(x_{\pi(k)}))$$

in R, where $k>1$, $q, q_1,\dots,q_k \in Q$, $\sigma \in \Sigma^{(k)}$, $\delta \in \Delta^{(k)}$, $\gamma, \gamma_1,\dots,\gamma_k \in (\Delta^{(1)})^*$ and $\pi : [k] \to [k]$ is a permutation such that q is maximal among the permutation states, i.e., there is no permutation state p with $p > q$.

Since μ is a permutation rule, there exist $n>1$, a sequence $1 \leqslant i_1,\dots,i_n \leqslant k$ of different indexes such that $i_2 = \pi(i_1),\dots,i_n = \pi(i_{n-1})$ and $i_1 = \pi(i_n)$. Moreover, since q is maximal among the permutation states and M is shape preserving, there is a derivation such that the occurrence of δ which is added to the output by the application of μ matches the occurrence of σ to which μ was applied. Hence the following statement holds.

(*) For every $1 \leqslant j \leqslant n$, a tree in $dom(\tau_{M,q_{\pi^{-1}(i_j)}})$ has the same shape as a tree in $\gamma_{i_j} ran(\tau_{M,q_{i_j}})$, see Fig. 4. Hence $dom(\tau_{M,q_{\pi^{-1}(i_j)}}) \cup \gamma_{i_j} ran(\tau_{M,q_{i_j}})$ is uniform.

Now let, for every $1 \leqslant j \leqslant n$, $M^{(j)} = (Q^{(j)}, \Sigma, \Delta, p^{(j)}, R^{(j)})$ be the top-down quasirelabeling which induces the relation $dom(\tau_{M,q_{\pi^{-1}(i_j)}}) \times ran(\tau_{M,q_{i_j}})$. Such a top-down quasirelabeling exists by Lemmas 3.11 and 3.12. Assume that the state sets $Q^{(j)}$ are disjoint.

Moreover, let $\bar{\mu}$ be the rule obtained from μ as follows. For every $1 \leqslant j \leqslant n$, we substitute the construct $q_{i_j}(x_{\pi(i_j)})$ by $p^{(j)}(x_{i_j})$ in the right-hand side of μ. Then the permutation degree of $\bar{\mu}$ is less than that of μ.

Now construct the top-down tree transducer $(\tilde{Q}, \Sigma, \Delta, \tilde{q_0}, \tilde{R})$, where

- $\tilde{q_0} = q_0$,
- $\tilde{Q} = Q \cup Q^{(1)} \cup \cdots \cup Q^{(n)}$,
- $\tilde{R} = (R - \{\mu\}) \cup \{\bar{\mu}\} \cup R^{(1)} \cup \cdots \cup R^{(n)}$.

Then eliminate the useless rules of the top-down tree transducer $(\tilde{Q}, \Sigma, \Delta, \tilde{q_0}, \tilde{R})$ and let the resulting top-down tree transducer be $N = (\bar{Q}, \Sigma, \Delta, \bar{q_0}, \bar{R})$. Then it should be clear that the permutation degree of N is less than that of M.

Now we show that $\tau_M = \tau_N$.

Let us denote $\Rightarrow_{M \cup N}$ the derivation relation over $T_{Q \cup \bar{Q} \cup \Sigma \cup \Delta}$ in which both the rules in R and in $\bar{R}$ can be applied.

First we show $\tau_M \subseteq \tau_N$. To see this, it is sufficient to show the following. Let $s \in T_\Sigma$ and $t \in T_\Delta$ with $q_0(s) \Rightarrow^*_{M \cup N} t$ and let a derivation sequence of t from $q_0(s)$ be given such that the rule μ is applied $K \geqslant 1$ times in the sequence. Then we can construct another derivation sequence from $q_0(s)$ to t such that the rule μ is applied $K-1$ times in the steps of the second derivation sequence.

Indeed, if $q_0(s) \Rightarrow^*_M t$ such that the rule μ is applied K times in the steps of that derivation, then applying K times the construction we obtain that $q_0(s) \Rightarrow^*_N t$.

Let us take a derivation $q_0(s) \Rightarrow^*_{M \cup N} t$, in which the rule μ is applied K times. Let $\rho : [n] \to [n]$ be a permutation such that $i_{\rho(1)} < \cdots < i_{\rho(n)}$. Then $q_0(s) \Rightarrow^*_{M \cup N} t$ can be written as

$$
\begin{aligned}
q_0(s) &= q_0(u[\sigma(s_1, \ldots, s_k)]) \\
&\Rightarrow^*_{M \cup N} v[q(\sigma(s_1, \ldots, s_k))] \\
&\Rightarrow_M v[\gamma\delta(\ldots, \gamma_{i_{\rho(1)}} q_{i_{\rho(1)}}(s_{\pi(i_{\rho(1)})}), \ldots, \gamma_{i_{\rho(n)}} q_{i_{\rho(n)}}(s_{\pi(i_{\rho(n)})}), \ldots)] \quad \text{(rule } \mu) \\
&\Rightarrow^*_{M \cup N} v[\gamma\delta(\ldots, \gamma_{i_{\rho(1)}} t_1, \ldots, \gamma_{i_{\rho(n)}} t_n, \ldots)] \\
&= t,
\end{aligned}
$$

where $u \in \widehat{T_\Sigma}(X_1)$, $s_1, \ldots, s_k \in T_\Sigma$, $v \in \widehat{T_\Delta}(X_1)$ and $t_1, \ldots, t_n \in T_\Delta$.
Then

$$
\begin{aligned}
q_0(s) &= q_0(u[\sigma(s_1, \ldots, s_k)]) \\
&\Rightarrow^*_{M \cup N} v[q(\sigma(s_1, \ldots, s_k))] \\
&\Rightarrow_N v[\gamma\delta(\ldots, \gamma_{i_{\rho(1)}} p^{(\rho(1))}(s_{i_{\rho(1)}}), \ldots, \gamma_{i_{\rho(n)}} p^{(\rho(n))}(s_{i_{\rho(n)}}), \ldots)] \quad \text{(rule } \bar{\mu}) \\
&\Rightarrow^*_{M \cup N} v[\gamma\delta(\ldots, \gamma_{i_{\rho(1)}} t_1, \ldots, \gamma_{i_{\rho(n)}} t_n, \ldots)] \\
&= t
\end{aligned}
$$

because, for every $1 \leqslant j \leqslant n$, $t_j \in ran(\tau_{M, q_{i_{\rho(j)}}})$ and $s_{i_{\rho(j)}} \in dom(\tau_{M, q_{\pi^{-1}(i_{\rho(j)})}})$ and $p^{(\rho(j))}$ is the initial state of the top-down quasirelabeling which induces the tree transformation $dom(\tau_{M, q_{\pi^{-1}(i_{\rho(j)})}}) \times ran(\tau_{M, q_{i_{\rho(j)}}})$.

The above argumentation is clearly reversible, so $\tau_N \subseteq \tau_M$ also holds. □

Now we are at the third part. Here we can show that every shape preserving top-down quasirelabeling is effectively equivalent to a relabeling. For this we need again some preparations.

Definition 3.18. Let Σ and Δ be ranked alphabets.
(a) We denote by $\langle\Sigma,\Delta\rangle$ the ranked alphabet defined by $\langle\Sigma,\Delta\rangle^{(k)} = \Sigma^{(k)} \times \Delta^{(k)}$, for every $k \geqslant 0$.
(b) We denote by $\Sigma_\diamondsuit$ and $\Delta_\diamondsuit$ the ranked alphabets which are obtained from Σ and Δ, respectively, by adding a new unary symbol $\diamondsuit$ to both $\Sigma^{(1)}$ and $\Delta^{(1)}$.
(c) We define the tree homomorphisms $h_\Sigma : T_{\langle\Sigma_\diamondsuit,\Delta_\diamondsuit\rangle} \to T_\Sigma$ and $h_\Delta : T_{\langle\Sigma_\diamondsuit,\Delta_\diamondsuit\rangle} \to T_\Delta$ in the following way. For every $k \geqslant 0$ and $\langle\sigma,\delta\rangle \in \langle\Sigma_\diamondsuit,\Delta_\diamondsuit\rangle^{(k)}$, let

$$h_\Sigma(\langle\sigma,\delta\rangle(x_1,\ldots,x_k)) = \begin{cases} \sigma(x_1,\ldots,x_k) & \text{if } \sigma \neq \diamondsuit \\ x_1 & \text{otherwise,} \end{cases}$$

$$h_\Delta(\langle\sigma,\delta\rangle(x_1,\ldots,x_k)) = \begin{cases} \delta(x_1,\ldots,x_k) & \text{if } \delta \neq \diamondsuit \\ x_1 & \text{otherwise.} \end{cases}$$

Notice that in this definition $\sigma = \diamondsuit$ (resp. $\delta = \diamondsuit$) implies $\delta \in \Delta_\diamondsuit^{(1)}$ (resp. $\sigma \in \Sigma_\diamondsuit^{(1)}$).

Observation 3.19. For every $s \in T_{\langle\Sigma,\Delta\rangle}$, $h_\Sigma(s) \approx h_\Delta(s)$. Hence, for every $L \subseteq T_{\langle\Sigma,\Delta\rangle}$, the tree transformation $h_\Sigma^{-1} \circ Id(L) \circ h_\Delta$ is shape preserving, where $Id(L) = \{(s,s) \mid s \in L\}$. The same does not hold with the ranked alphabet $\langle\Sigma_\diamondsuit,\Delta_\diamondsuit\rangle$.

Lemma 3.20. *For every top-down quasirelabeling $M = (Q,\Sigma,\Delta,q_0,R)$ a top-down tree automaton $T = (P,\langle\Sigma_\diamondsuit,\Delta_\diamondsuit\rangle,p_0,R_T)$ can be constructed such that $\tau_M = h_\Sigma^{-1} \circ \tau_T \circ h_\Delta$.*

Proof. For each rule

$$\mu : q(\sigma(x_1,\ldots,x_k)) \to \gamma\delta(\gamma_1 q_1(x_1),\ldots,\gamma_k q_k(x_k)) \in R,$$

where $k \neq 1$ $q,q_1,\ldots,q_k \in Q$, $\sigma \in \Sigma^{(k)}$, $\delta \in \Delta^{(k)}$, and $\gamma,\gamma_1,\ldots,\gamma_k \in (\Delta^{(1)})^*$ with $length(\gamma) = n$ and $length(\gamma_1) = n_1,\ldots,length(\gamma_k) = n_k$, construct the rules

$$\begin{aligned} &q(\langle\diamondsuit,\gamma(1)\rangle(x_1)) && \to \langle\diamondsuit,\gamma(1)\rangle(p_1(x_1)), \\ &\ldots \\ &p_{n-1}(\langle\diamondsuit,\gamma(n)\rangle(x_1)) && \to \langle\diamondsuit,\gamma(n)\rangle(p_n(x_1)), \\ &p_n(\langle\sigma,\delta\rangle(x_1,\ldots,x_k)) && \to \langle\sigma,\delta\rangle(p_{11}(x_1),\ldots,p_{k1}(x_k)), \qquad (\dagger) \end{aligned}$$

as well as, for every $1 \leqslant j \leqslant k$, construct the rules

$$p_{j1}(\langle\Diamond, \gamma_j(1)\rangle(x_1)) \rightarrow \langle\Diamond, \gamma_j(1)\rangle(p_{j2}(x_1)),$$
$$\ldots$$
$$p_{jn_j}(\langle\Diamond, \gamma_j(n_j)\rangle(x_1)) \rightarrow \langle\Diamond, \gamma_j(n_j)\rangle(q_j(x_1)),$$

where $p_1, \ldots, p_n$ and $p_{j1}, \ldots, p_{jn_j}$ are new states. (In case $n = 0$ we mean $p_n = q$ in the rule (†) and there are no rules above (†). In case $n_j = 1$, we have the only rule $p_{j1}(\langle\Diamond, \gamma_j(1)\rangle(x_1)) \rightarrow \langle\Diamond, \gamma_j(1)\rangle(q_j(x_1))$, while in case $n_j = 0$, p_{j1} is meant to be q_j in the rule (†) and we do not need further rules for the index j.)

Let R_μ be the set of all rules constructed from μ and Q_μ be the set of all states appearing in those rules.

Moreover, for each rule $\mu : q(\sigma(x_1)) \rightarrow \gamma p(x_1)$ in R, where $q, p \in Q$, $\sigma \in \Sigma^{(1)}$ and $\gamma \in (\Delta^{(1)})^*$ with $length(\gamma) = n$, construct the rules

$$q(\langle\Diamond, \gamma(1)\rangle(x_1)) \rightarrow \langle\Diamond, \gamma(1)\rangle(p_1(x_1)),$$
$$\ldots$$
$$p_{n-1}(\langle\sigma, \gamma(n)\rangle(x_1)) \rightarrow \langle\sigma, \gamma(n)\rangle(p(x_1)),$$

where $p_1, \ldots, p_{n-1}$ are new states. (In case $n = 1$, we have the only rule $q(\langle\sigma, \gamma(1)\rangle(x_1)) \rightarrow \langle\sigma, \gamma(1)\rangle(p(x_1))$, while in case $n = 0$, we have the only rule $q(\langle\sigma, \Diamond\rangle(x_1)) \rightarrow \langle\sigma, \Diamond\rangle(p(x_1))$.) Again, let R_μ be the set of all rules constructed from μ and Q_μ be the set of all states appearing in those rules.

Now let $T = (P, \langle\Sigma_\Diamond, \Delta_\Diamond\rangle, p_0, R_T)$ be the top-down tree automaton, where $P = \bigcup_{\mu \in R} Q_\mu$, $p_0 = q_0$ and $R_T = \bigcup_{\mu \in R} R_\mu$. It should be clear that $\tau_M = h_\Sigma^{-1} \circ \tau_T \circ h_\Delta$. □

Now we define the set of segments of a tree. The corresponding notion for strings was introduced in [7]. Broadly speaking it consists of those parts of the tree which are built up from unary symbols.

Definition 3.21. Let Σ and Δ be ranked alphabets and $v \in T_{\langle\Sigma_\Diamond, \Delta_\Diamond\rangle}$. Let

$$F(v) = \{\gamma \in (\langle\Sigma_\Diamond, \Delta_\Diamond\rangle^{(1)})^+ \mid \text{there exist } s \in \widehat{T}_{\langle\Sigma_\Diamond,\Delta_\Diamond\rangle}(X_1) \text{ and } u \in T_{\langle\Sigma_\Diamond,\Delta_\Diamond\rangle} \text{ such that } s[\gamma u] = v\}.$$

A $\gamma \in F(v)$ is *maximal* if $s[\gamma u] = v$ and, for every $s' \in \hat{T}_{\langle\Sigma_\Diamond,\Delta_\Diamond\rangle}(X_1)$, $\gamma' \in (\langle\Sigma_\Diamond, \Delta_\Diamond\rangle^{(1)})^*$ and $u' \in T_{\langle\Sigma_\Diamond,\Delta_\Diamond\rangle}$, if $occ(s', x_1)$ is a prefix of $occ(s, x_1)$ and u' is a subtree of u, and $s'[\gamma' u'] = v$, then $s = s'$, $\gamma = \gamma'$ and $u = u'$.

For a tree language $L \subseteq T_{\langle\Sigma_\Diamond,\Delta_\Diamond\rangle}$, we define $F(L) = \bigcup_{v \in L} F(v)$. The tree language L is k-bounded, where $k \geqslant 0$ is an integer if, for every $\gamma \in F(L)$, the approximation $|length(h_\Sigma(\gamma)) - length(h_\Delta(\gamma))| \leqslant k$ holds. Moreover, L is bounded if it is k-bounded for some k.

Now we prove two technical lemmas. The proof of both rely on the simple idea that some state of a finite tree automaton necessarily repeats on sufficiently long paths (and thus segments) of a tree.

Lemma 3.22. *Let Σ and Δ be ranked alphabets and $T=(P,\langle\Sigma_\diamond,\Delta_\diamond\rangle,p_0,R_T)$ be a top-down tree automaton such that the tree transformation $\tau=h_\Sigma^{-1}\circ\tau_T\circ h_\Delta$ is shape preserving. Then $L(T)$ is bounded.*

Proof. We prove by contradiction that $L(T)$ is k-bounded, where $k=\|P\|$. Assume that $L(T)$ is not k-bounded, let

$$L_k=\{\gamma\in F(L(T))\mid |\mathit{length}(h_\Sigma(\gamma))-\mathit{length}(h_\Delta(\gamma))|>k\}$$

and let γ be an element of L_k with minimal length. Then also $\mathit{length}(\gamma)>k$. Moreover, for every $0\leqslant i\leqslant \mathit{length}(\gamma)$, let $\beta_i,\gamma_i\in(\langle\Sigma_\diamond,\Delta_\diamond\rangle^{(1)})^*$ be such that $\mathit{length}(\beta_i)=i$ and $\gamma=\beta_i\gamma_i$.

Since $\gamma\in F(L(T))$ and $\mathit{length}(\gamma)>k$, there are $s\in\hat{T}_{\langle\Sigma_\diamond,\Delta_\diamond\rangle}(X_1)$, $u\in T_{\langle\Sigma_\diamond,\Delta_\diamond\rangle}$, q, $q'\in P$ and $0\leqslant i<j\leqslant \mathit{length}(\gamma)$ such that $s[\gamma u]\in L(T)$ and

$$p_0(s[\gamma u])\Rightarrow_T^* s[q(\gamma u)]\Rightarrow_T^* s[\beta_i q'(\gamma_i u)]\Rightarrow_T^* s[\beta_j q'(\gamma_j u)]\Rightarrow_T^* s[\gamma u].$$

Let $\beta_{ij}\in(\langle\Sigma_\diamond,\Delta_\diamond\rangle^{(1)})^+$ be such that $\beta_i\beta_{ij}=\beta_j$ and thus $\beta_i\beta_{ij}\gamma_j=\gamma$. Then, for every $l\geqslant 0$, $s[\beta_i\beta_{ij}^l\gamma_j u]\in L(T)$ hence $\beta_i\beta_{ij}^l\gamma_j\in F(L(T))$.

Now $\mathit{length}(h_\Sigma(\beta_{ij}))\neq \mathit{length}(h_\Delta(\beta_{ij}))$ because otherwise the tree $\gamma'=\beta_i\gamma_j\in F(L(T))$ also satisfies $|\mathit{length}(h_\Sigma(\gamma'))-\mathit{length}(h_\Delta(\gamma'))|>k$ thus also $\gamma'\in L_k$. This, however, is impossible because $\mathit{length}(\gamma')<\mathit{length}(\gamma)$ and γ is an element of L_k with minimal length.

Thus we can assume that, say, $\mathit{length}(h_\Sigma(\beta_{ij}))>\mathit{length}(h_\Delta(\beta_{ij}))$. Then, for a sufficiently big l,

$$\mathit{height}(h_\Sigma(s[\beta_i\beta_{ij}^l\gamma_j u]))>\mathit{height}(h_\Delta(s[\beta_i\beta_{ij}^l\gamma_j u])),$$

which contradicts the fact, that τ is shape preserving. Hence $L(T)$ is k-bounded. $\square$

We will also need the following result.

Lemma 3.23. *Let Σ and Δ be ranked alphabets and $T=(P,\langle\Sigma_\diamond,\Delta_\diamond\rangle,p_0,R_T)$ be a top-down tree automaton. Then it is decidable if $L(T)$ is bounded and, moreover, if $L(T)$ is bounded, then we can compute the smallest k for which $L(T)$ is k-bounded.*

Proof. Let $n=\|P\|$ and define for every $p,q\in P$

$$L_{p,q}^{(n)}=\{\gamma\in(\langle\Sigma_\diamond,\Delta_\diamond\rangle^{(1)})^*\mid p(\gamma)\Rightarrow_T^*\gamma q(x_1)\text{ and }\mathit{length}(\gamma)\leqslant n\}.$$

It should be clear that $L_{p,q}^{(n)}$ is a finite and computable set.

(a) First we show that it is decidable whether $L(T)$ is bounded. To see this we prove that $L(T)$ is bounded if and only if, for every $p\in P$ and $\gamma\in L_{p,p}^{(n)}$ the condition $|\mathit{length}(h_\Sigma(\gamma))-\mathit{length}(h_\Delta(\gamma))|=0$ holds.

Assume that $L(T)$ is bounded and there are $p \in P$ and $\gamma \in L^{(n)}_{p,p}$ such that $|length(h_\Sigma(\gamma)) - length(h_\Delta(\gamma))| > 0$. Then, since all states in p are useful, there are $s \in \hat{T}_{\langle \Sigma_\diamond, \Delta_\diamond \rangle}(X_1)$ and $u \in T_{\langle \Sigma_\diamond, \Delta_\diamond \rangle}$ such that $s[\gamma u] \in L(T)$, hence $\gamma \in F(L(T))$. Since $p(\gamma) \Rightarrow^*_T \gamma\, p(x_1)$, for every $l > 0$, $s[\gamma^l u] \in L(T)$ and thus $\gamma^l \in F(L(T))$. This means that $L(T)$ is not bounded, which is a contradiction.

Next assume that $L(T)$ is not bounded. Let

$$L_n = \{\gamma \in F(L(T)) \mid |length(h_\Sigma(\gamma)) - length(h_\Delta(\gamma))| > n\}$$

and let γ be and element of L_n with minimal length. Then certainly $length(\gamma) > n$ and thus there are $\gamma_1, \gamma_2, \gamma_3 \in (\langle \Sigma_\diamond, \Delta_\diamond \rangle^{(1)})^*$ such that $0 < length(\gamma_2) \leqslant n$ and $\gamma = \gamma_1\gamma_2\gamma_3$, moreover, there are states $p, q \in P$ such that $q(\gamma_1\gamma_2\gamma_3) \Rightarrow^*_T \gamma_1\, p(\gamma_2\gamma_3) \Rightarrow^*_T \gamma_1\gamma_2\, p(\gamma_3)$. Then $\gamma_2 \in L^{(n)}_{p,p}$ and $\gamma_1\gamma_3 \in F(L(T))$. Now $|length(h_\Sigma(\gamma_2)) - length(h_\Delta(\gamma_2))| > 0$. Indeed, if $|length(h_\Sigma(\gamma_2)) - length(h_\Delta(\gamma_2))| = 0$, then $|length(h_\Sigma(\gamma_1\gamma_3)) - length(h_\Delta(\gamma_1\gamma_3))| > n$ and thus $\gamma_1\gamma_3 \in L_n$, which is contradiction because $length(\gamma_1\gamma_3) < length(\gamma)$.

Now since it is decidable if, for every $p \in P$ and $\gamma \in L^{(n)}_{p,p}$ the condition $|length(h_\Sigma(\gamma)) - length(h_\Delta(\gamma))| = 0$ holds, it is also decidable if $L(T)$ is bounded.

(b) Now assume that $L(T)$ is bounded. Let

$$k = \max\{|length(h_\Sigma(\gamma)) - length(h_\Delta(\gamma))| \mid \gamma \in L^{(n)}_{p,q},\ p, q \in Q\}.$$

Certainly there is no $k' < k$ such that $L(T)$ is k'-bounded. On the other hand, we can show that $L(T)$ is k-bounded.

Let us suppose that $L(T)$ is not k-bounded and let

$$L_k = \{\gamma \in F(L(T)) \mid |length(h_\Sigma(\gamma)) - length(h_\Delta(\gamma))| > k\}$$

and γ be an element of L_k with minimal length. Now, by the definition of k, $length(\gamma) > n$ and thus there are $\gamma_1, \gamma_2, \gamma_3 \in (\langle \Sigma_\diamond, \Delta_\diamond \rangle^{(1)})^*$ and states $p, q \in P$ with the same properties as in case (a). Then $|length(h_\Sigma(\gamma_2)) - length(h_\Delta(\gamma_2))| = 0$, because $L(T)$ is bounded, see case (a). Moreover, $|length(h_\Sigma(\gamma_1\gamma_3)) - length(h_\Delta(\gamma_1\gamma_3))| > k$, hence $\gamma_1\gamma_3 \in L_k$. This is a contradiction because $length(\gamma_1\gamma_3) < length(\gamma)$ and γ is an element of L_k with minimal length. Hence $L(T)$ is k-bounded. □

The following definition is a key to constructing the relabeling equivalent to a shape preserving top-down quasirelabeling.

Definition 3.24. Let $T = (P, \langle \Sigma_\diamond, \Delta_\diamond \rangle, q_0, R_T)$ be a top-down tree automaton such that $L(T)$ is k-bounded. The *shape preserving frame of* T is the top-down tree automaton $T' = (P', \langle \Sigma, \Delta \rangle, p'_0, R_{T'})$ constructed in the following way.

(a) First we construct a linear deterministic top-down tree transducer $N = (P_N, \langle \Sigma_\diamond, \Delta_\diamond \rangle, \langle \Sigma, \Delta \rangle, p_0, R_N)$ as follows. Let

- $P_N = (\Sigma^{(1)})^{*,k} \times \{\varepsilon\} \cup \{\varepsilon\} \times (\Delta^{(1)})^{*,k}$,
- $p_0 = [\varepsilon, \varepsilon]$, (We use the brackets [and] to denote elements of P_N.)
- R_N is the smallest set of rules satisfying the following conditions.
  - For every m with $m \neq 1$, $\sigma \in \Sigma^{(m)}$ and $\delta \in \Delta^{(m)}$, the rule $[\varepsilon, \varepsilon](\langle \sigma, \delta \rangle(x_1, \ldots, x_m)) \to \langle \sigma, \delta \rangle([\varepsilon, \varepsilon](x_1), \ldots, [\varepsilon, \varepsilon](x_m))$ is in R_N.

- For every $u \in (\Sigma^{(1)})^{*,k-1}$ and $\beta \in \Sigma^{(1)}$, the rule $[u,\varepsilon](\langle\beta,\diamondsuit\rangle(x_1)) \to [u\beta,\varepsilon](x_1)$ is in R_N.
- For every $\beta,\beta' \in \Sigma^{(1)}$, $u \in (\Sigma^{(1)})^{*,k-1}$ and $\gamma \in \Delta^{(1)}$, the rule $[\beta' u,\varepsilon](\langle\beta,\gamma\rangle(x_1)) \to \langle\beta',\gamma\rangle[u\beta,\varepsilon](x_1)$ is in R_N.
- For every $\beta \in \Sigma^{(1)}$, $u \in (\Sigma^{(1)})^{*,k-1}$ and $\gamma \in \Delta^{(1)}$, the rule $[\beta u,\varepsilon](\langle\diamondsuit,\gamma\rangle(x_1)) \to \langle\beta,\gamma\rangle[u,\varepsilon](x_1)$ is in R_N.
- For every $v \in (\Delta^{(1)})^{*,k-1}$ and $\gamma \in \Delta^{(1)}$, the rule $[\varepsilon,v](\langle\diamondsuit,\gamma\rangle(x_1)) \to [\varepsilon,v\gamma](x_1)$ is in R_N.
- For every $\gamma,\gamma' \in \Delta^{(1)}$, $v \in (\Delta^{(1)})^{*,k-1}$ and $\beta \in \Sigma^{(1)}$, the rule $[\varepsilon,\gamma' v](\langle\beta,\gamma\rangle(x_1)) \to \langle\beta,\gamma'\rangle[\varepsilon,v\gamma](x_1)$ is in R_N.
- For every $\gamma' \in \Delta^{(1)}$, $v \in (\Delta^{(1)})^{*,k-1}$ and $\beta \in \Sigma^{(1)}$, the rule $[\varepsilon,\gamma' v](\langle\beta,\diamondsuit\rangle(x_1)) \to \langle\beta,\gamma'\rangle[\varepsilon,v](x_1)$ is in R_N.

Now let $L = ran(\tau_T \circ \tau_N)$. Since linear top-down tree transducers preserve recognizability, [4] IV. 6.6 Corollary, L is recognizable, hence a top-down tree automaton $T' = (P', \langle\Sigma,\Delta\rangle, p'_0, R_{T'})$ can be constructed such that $L = dom(\tau_{T'})$.

Now we prove the following lemma which hopefully provides justification for the definition of the shape preserving frame.

Lemma 3.25. *Let $T = (P, \langle\Sigma_\diamondsuit,\Delta_\diamondsuit\rangle, q_0, R_T)$ be a top-down tree automaton such that $L(T)$ is k-bounded and $T' = (P', \langle\Sigma,\Delta\rangle, p'_0, R_{T'})$ is the shape preserving frame of T. Then $h_\Sigma^{-1} \circ \tau_{T'} \circ h_\Delta$ is shape preserving and $h_\Sigma^{-1} \circ \tau_{T'} \circ h_\Delta \subseteq h_\Sigma^{-1} \circ \tau_T \circ h_\Delta$.*

Proof. Since T' recognizes trees over the ranked alphabet $\langle\Sigma,\Delta\rangle$, by Observation 3.19, the tree transformation $h_\Sigma^{-1} \circ \tau_{T'} \circ h_\Delta$ is shape preserving.

Since $L(T)$ is k-bounded, for every maximal $\gamma \in F(L(T))$ and prefix γ' of γ, the approximation $|length(h_\Sigma(\gamma')) - length(h_\Delta(\gamma'))| \leqslant k$ holds. Now the piece of string of length at most k with which $h_\Sigma(\gamma')$ is "ahead" of $h_\Delta(\gamma')$ ($h_\Delta(\gamma')$ is ahead of $h_\Sigma(\gamma')$) is stored in the first (second) component of the states of N.

Moreover, N is able to process an input symbol $\langle\sigma,\delta\rangle \in \langle\Sigma_\diamondsuit,\Delta_\diamondsuit\rangle^{(m)}$ with $m \neq 1$ only in state $[\varepsilon,\varepsilon]$. Hence it should be clear that N has the following property. For every $v \in ran(\tau_T)$, the inclusion $v \in dom(\tau_N)$ holds, if and only if, for every maximal $\gamma \in F(v)$, $|length(h_\Sigma(\gamma)) - length(h_\Delta(\gamma))| = 0$. Moreover, if this is the case, then $h_\Sigma(v) = h_\Sigma(v')$ and $h_\Delta(v) = h_\Delta(v')$, where $v' = \tau_N(v)$.

Now we can show that $h_\Sigma^{-1} \circ \tau_{T'} \circ h_\Delta \subseteq h_\Sigma^{-1} \circ \tau_T \circ h_\Delta$ in the following way. Let $(s,t) \in h_\Sigma^{-1} \circ \tau_{T'} \circ h_\Delta$. Then, there is a $v \in ran(\tau_T) \cap dom(\tau_N)$ such that, for $v' = \tau_N(v)$, we have $h_\Sigma(v') = s$ and $h_\Delta(v') = t$. Since $v \in ran(\tau_T) \cap dom(\tau_N)$, by the above note $h_\Sigma(v) = h_\Sigma(v')$ and $h_\Delta(v) = h_\Delta(v')$. Hence, also $(s,t) \in h_\Sigma^{-1} \circ \tau_T \circ h_\Delta$. $\square$

The following also holds.

Lemma 3.26. *Let $T = (P, \langle\Sigma_\diamondsuit,\Delta_\diamondsuit\rangle, q_0, R_T)$ be a top-down tree automaton such that $L(T)$ is k-bounded and $T' = (P', \langle\Sigma,\Delta\rangle, p'_0, R_{T'})$ is the shape preserving frame of T. If $h_\Sigma^{-1} \circ \tau_T \circ h_\Delta$ is shape preserving, then $h_\Sigma^{-1} \circ \tau_T \circ h_\Delta \subseteq h_\Sigma^{-1} \circ \tau_{T'} \circ h_\Delta$.*

Proof. Let $(s,t)\in h_\Sigma^{-1}\circ\tau_T\circ h_\Delta$, then there is a $v\in ran(\tau_T)$ such that $h_\Sigma(v)=s$ and $h_\Delta(v)=t$. Since $s\approx t$, for every maximal $\gamma\in F(v)$, $|length(h_\Sigma(\gamma))-length(h_\Delta(\gamma))|=0$ holds. Then, by the proof of Lemma 3.25, $v\in dom(\tau_N)$ and $h_\Sigma(v)=h_\Sigma(v')$ and $h_\Delta(v)=h_\Delta(v')$, where $v'=\tau_N(v)$. This means $(s,t)\in h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$. □

Corollary 3.27. *Let $T=(P,\langle\Sigma_\diamond,\Delta_\diamond\rangle,q_0,R_T)$ be a top-down tree automaton such that $L(T)$ is k-bounded and $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta$ is shape preserving. Then $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta=h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$, where $T'=(P',\langle\Sigma,\Delta\rangle,p_0',R_{T'})$ is the shape preserving frame of T.*

Proof. It immediately follows from Lemmas 3.25 and 3.26. □

In order to state also decidability of the shape preserving property, we will need the following result.

Lemma 3.28. *Let $T=(P,\langle\Sigma_\diamond,\Delta_\diamond\rangle,q_0,R_T)$ be a top-down tree automaton such that $L(T)$ is k-bounded and $T'=(P',\langle\Sigma,\Delta\rangle,p_0',R_{T'})$ the shape preserving frame of T. Then it is decidable, if $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta\subseteq h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$ holds.*

Proof. We show that $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta\subseteq h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$ if and only if $ran(\tau_T)\subseteq dom(\tau_N)$.

Assume that $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta\subseteq h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$ and let $v\in ran(\tau_T)$. Let $s=h_\Sigma(v)$ and $t=h_\Delta(v)$, then $(s,t)\in h_\Sigma^{-1}\circ\tau_T\circ h_\Delta$ and thus $(s,t)\in h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$. Then, there is a $v'\in ran(\tau_T\circ\tau_N)$ such that $s=h_\Sigma(v')$ and $t=h_\Delta(v')$. Since $v'\in T_{\langle\Sigma,\Delta\rangle}$, by Observation 3.19, $s\approx t$ holds, which implies that for every maximal $\gamma\in F(v)$, $|length(h_\Sigma(\gamma))-length(h_\Delta(\gamma))|=0$ holds. Consequently, $v\in dom(\tau_N)$.

Now assume that $ran(\tau_T)\subseteq dom(\tau_N)$ and let $(s,t)\in h_\Sigma^{-1}\circ\tau_T\circ h_\Delta$. Then, there is a $v\in ran(\tau_T)$ such that $s=h_\Sigma(v)$ and $t=h_\Delta(v)$. Now $v\in dom(\tau_N)$ also holds, which implies that there is a $v'\in ran(\tau_N)$ such that $\tau_N(v)=v'$. It follows from the construction of N that $s=h_\Sigma(v')$ and $t=h_\Delta(v')$. Then $(s,t)\in h_\Sigma^{-1}\circ(\tau_T\circ\tau_N)\circ h_\Delta$, cf. the proof of Lemma 3.25. Consequently $(s,t)\in h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$.

Then the decidability of $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta\subseteq h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$ follows from the fact that both $ran(\tau_T)$ and $dom(\tau_N)$ are recognizable tree languages and that the inclusion problem is decidable for recognizable tree languages (Theorem 10.3 in Chapter II. of 4). □

As the last step of the preparation we state an obvious fact.

Lemma 3.29. *Let $T=(P,\langle\Sigma,\Delta\rangle,p_0,R_T)$ be a top-down tree automaton. Then there is a relabeling $M=(P,\Sigma,\Delta,p_0,R_M)$ such that $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta=\tau_M$.*

Proof. R_M is constructed as follows. For each rule $q(\langle\sigma,\delta\rangle(x_1,\dots,x_k))\to\langle\sigma,\delta\rangle(q_1(x_1),\dots,q_k(x_k))$ in R_T, let the rule $q(\sigma(x_1,\dots,x_k))\to\delta(q_1(x_1),\dots,q_k(x_k))$ in R_M. It should be clear that $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta=\tau_M$. □

Now we can state one of our main result.

Theorem 3.30. *Every shape preserving top-down tree transducer is equivalent to a relabeling tree transducer.*

Proof. It follows form Lemmas 3.7, 3.17, 3.20, 3.22, Corollary 3.27 and Lemma 3.29. □

Our other main result is the following.

Theorem 3.31. *It is decidable if a top-down tree transducer* $M=(Q,\Sigma,\Delta,q_0,R)$ *is shape preserving or not.*

Proof. We can assume without loss of generality that each rule in R is useful. We give an algorithm that terminates with yes if M is shape preserving, otherwise it terminates with no. The algorithm is as follows.

1. Check if M is permutation top-down quasirelabeling or not. If not, then halt with no because, by Lemma 3.7, M is not shape preserving.
2. Compute the relation $>$ on Q (Definition 3.13) according to Lemma 3.15. If $>$ contains a cycle, then halt with no, because, by Lemma 3.16, M is not shape preserving.
3. Eliminate the permutation rules from M in the following way (cf. Lemma 3.17). Take a permutation rule μ with a maximal state in its left-hand side with respect to $>$. Check, if the condition $(*)$ described in Lemma 3.17 holds. (Note that the tree languages $dom(\tau_{M,q_{\pi^{-1}(i_j)}})$ and $\gamma_{i_j}ran(\tau_{M,q_{i_j}})$ are recognizable, and thus are effectively computable.) If the condition $(*)$ does not hold, then halt with no, because M is not shape preserving. Otherwise eliminate the rule μ. (Finally M becomes a top-down quasirelabeling.)
4. Then, by Lemma 3.20, construct the top-down tree automaton $T=(P,\langle\Sigma_\diamond,\Delta_\diamond\rangle, p_0,R_T)$ such that $\tau_M=h_\Sigma^{-1}\circ\tau_T\circ h_\Delta$.
5. Check if $L(T)$ is bounded (Lemma 3.23). If not, then halt with no, because, by Lemma 3.22, M is not shape preserving. If yes, then compute k such that $L(T)$ is k-bounded (Lemma 3.23).
6. Construct the shape preserving frame $T'=(P',\langle\Sigma,\Delta\rangle,p_0',R_{T'})$ of T (Definition 3.24).
7. Check, if $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta\subseteq h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$ (Lemma 3.28). If not, then halt with no, because, by Lemma 3.25, M is not shape preserving.
8. Halt with yes (by Corollary 3.27, $h_\Sigma^{-1}\circ\tau_T\circ h_\Delta=h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$ and, by Lemma 3.29, $h_\Sigma^{-1}\circ\tau_{T'}\circ h_\Delta$ can be induced by a relabeling). □

4. Conclusions and further problems

We have shown that a top-down tree transducer is shape preserving if and only if it is equivalent to a relabeling tree transducer.

A corollary of this fact is that the class of relabeling tree transformations can be replaced by the class of shape preserving top-down tree transformations in several results concerning tree transducers. In what follows we give some concrete instances of this general observation.

First we recall the important and frequently used result from the theory of tree transducers that linear tree transformations preserve recognizability of tree languages

[8,2]. Now, since relabeling tree transducers are also linear, it easily follows from Theorem 3.30 (in fact already from Lemma 3.7) that shape preserving top-down tree transducers also preserve recognizability.

Another result is that $BOT = QREL \circ HOM$, which expresses that the class of bottom-up tree transformations is the same as the composition of the classes of relabeling tree transformations and the homomorphism tree transformations, for details see [2]. By Theorem 3.30 the class $QREL$ in this equation can be replaced by the class of shape preserving top-down tree transformations.

It would be nice to give a similar characterization of shape preserving bottom-up tree transducers [2]. We guess that the same holds, i.e., that every shape preserving bottom-up tree transducer is equivalent to a relabeling, however we could not prove this. Since linear and nondeleting bottom-up tree transformations are the same as linear and nondeleting top-down tree transformations [2], it would be sufficient to show that every shape preserving bottom-up tree transducer is equivalent to a linear and nondeleting bottom-up tree transducer, because then Theorem 3.30 would lead us to the guessed characterization of the shape preserving bottom-up tree transducers. Another way can be to use the mentioned equation $BOT = QREL \circ HOM$. Since a relabeling is shape preserving and its range is recognizable [2], this equation says that it is sufficient to consider whether the restriction $\tau_H|_L$ of a homomorphism tree transducer H to a recognizable tree language L is shape preserving.

Acknowledgements

The authors are grateful to Björn Borchardt for his valuable comments on a previous version of this paper.

References

[1] S. Eilenberg, Automata, Languages and Machines, Vol. A. Academic Press, New York, 1974.
[2] J. Engelfriet, Bottom-up and top-down tree transformations—a comparison, Math. Systems Theory 9 (1975) 198–231.
[3] Z. Fülöp, A. Terlutte, Iterated relabeling tree transducers, Theoret. Comput. Sci. 276 (2002) 221–244.
[4] F. Gécseg, M. Steinby, Tree Automata, Akadémiai Kiadó, Budapest, 1984.
[5] F. Gécseg, M. Steinby, Tree languages, in: G. Rozenberg, A. Salomaa (Eds.). Handbook of Formal Languages, Vol. 3, Springer-Verlag, Berlin, 1997, pp. 1–68.
[6] M. Latteux, D. Simplot, A. Terlutte, Iterated length-preserving rational transductions, in: Proc. MFCS 98, Vol. 1450 of Lecture Notes in Computer Science, Springer Verlag, Berlin, 1998, pp. 286–295.
[7] J. Leguy, Transductions rationelles décroissantes, R.A.I.R.O. Theoretical Informatics 5 (1981) 141–148.
[8] W.C. Rounds, Mappings and grammars on trees, Math. Systems Theory 4 (1970) 257–287.